Alcatel·Lucent

1675 LAMBDAUNITE MULTISERVICE SWITCH (MSS) | RELEASE 11.0

TL1 COMMAND GUIDE

TL1CG
365-374-199R11.0
CC109671784
ISSUE 1
JULY 2008

Contents

3 TL1 commands - alphabetical order

List of tables

About this document

Purpose

This document describes the external TL1 interface for 1675 LambdaUnite MSS in terms of TL1 command, responses, and notification definitions.

This document does not cover the specification of the TL1 protocol itself or that of underlying protocols of the OSI or IP stack.

Overview of the document

This manual is structured as follows:

- Chapter 1 contains introductory information (TL1 message index and command state diagram) and data type definitions, used to define parameters in TL1 commands or notifications
- Chapter 2 contains AID definitions, used to address entities with TL1 commands or notifications
- Chapter 3 contains the detailed requirements for the TL1 commands listed in alphabetical order
- Chapter 4 to chapter 13 contain the TL1 commands separated by the functional areas:
  - Administration and security
  - Configuration management
  - Fault management
  - Transmission and cross-connect handling
  - Protection
  - Performance monitoring
  - Synchronization
  - Software and database management
  - Data communications network
  - Optical network navigation system (ONNS)

- Chapter 14 contains a list of error codes and links to the corresponding TL1 commands
- Chapter 15 contains a list of TL1 commands and the corresponding user privilege codes in alphabetical order
- Chapter 16 contains a list of amendments related to the current software version

Supported systems

This manual describes 1675 LambdaUnite MSS, Release 11.0. For technical reasons, it is possible that features have been documented that are not available until later software versions. For precise information about the availability of features, consult the Software Release Description. This manual provides details of the status at the time of software delivery.

Intended use

This equipment shall be used only in accordance with intended use, corresponding installation, and maintenance statements as specified in this documentation. Any other use or modification is prohibited.

Optical safety

IEC customer laser safety guidelines

Alcatel-Lucent declares that this product is compliant with all essential safety requirements as stated in IEC 60825-Part 1 and 2 “Safety of laser products” and “Safety of optical fibre telecommunication systems”. Futhermore Alcatel-Lucent declares that the warning statements on labels on this equipment are in accordance with the specified laser radiation class.

Optical safety declaration

Alcatel-Lucent declares that this product is compliant with all essential safety requirements as stated in IEC 60825-Part 1 and 2 “Safety of Laser Products” and “Safety of Optical Fiber Telecommunication Systems”. Furthermore Alcatel-Lucent declares that the warning statements on labels on this equipment are in accordance with the specified laser radiation class.

Optical fiber communications

This equipment contains an Optical Fiber Communications semiconductor laser/LED transmitter. The following Laser Safety Guidelines are provided for this product.

General laser information

Optical fiber telecommunication systems, their associated test sets, and similar operating systems use semiconductor laser transmitters that emit infrared (IR) light at wavelengths between about 800 nanometers (nm) and 1600 nm. The emitted light is above the red end of the visible spectrum, which is normally not visible to the human eye. Although radiant en at near-IR wavelengths is officially designated invisible, some people can see the shorter wavelength energy even at power levels several orders of magnitude below any that have been shown to cause injury to the eye.

Conventional lasers can produce an intense beam of monochromatic light. The term “monochromaticity” means a single wavelength output of pure color that may be visible or invisible to the eye. A conventional laser produces a small-size beam of light, and because the beam size is small the power density (also called irradiance) is very high. Consequently, lasers and laser products are subject to federal and applicable state regulations, as well as international standards, for their safe operation.

A conventional laser beam expands very little over distance, or is said to be very well collimated. Thus, conventional laser irradiance remains relatively constant over distance. However, lasers used in light wave systems have a large beam divergence, typically 10 to 20 degrees. Here, irradiance obeys the inverse square law (doubling the distance reduces the irradiance by a factor of 4) and rapidly decreases over distance.

Lasers and eye damage

The optical energy emitted by laser and high-radiance LEDs in the 400-1400 nm range may cause eye damage if absorbed by the retina. When a beam of light enters the eye, the eye magnifies and focuses the energy on the retina magnifying the irradiance. The irradiance of the energy that reaches the retina is about 105, or 100,000 times more than at the cornea and, if sufficiently intense, may cause a retinal burn.

The damage mechanism at the wavelengths used in an optical fiber telecommunications is thermal in origin, i.e., damage caused by heating. Therefore, a specific amount of energy is required for a definite time to heat an area of retinal tissue. Damage to the retina occurs only when one looks at the light long enough that the product of the retinal irradiance and the viewing time exceeds the damage threshold. Optical energies above 1400 nm cause corneal and skin burns, but do not affect the retina. The thresholds for injury at wavelengths greater than 1400 nm are significantly higher than for wavelengths in the retinal hazard region.

Classification of lasers

Manufacturers of lasers and laser products in the U.S. are regulated by the Food and Drug Administration’s Center for Devices and Radiological Health (FDA/CDRH) under 21 CFR 1040. These regulations require manufacturers to certify each laser or laser product as belonging to one of four major Classes: I, II, lla, IlIa, lllb, or IV. The International Electro-technical Commission is an international standards body that writes laser safety standards under IEC-60825. Classification schemes are similar with Classes divided into Classes 1, 1M, 2, 2M, 3R, 3B, and 4. Lasers are classified according to the accessible emission limits and their potential for causing injury. Optical fiber telecommunication systems are generally classified as Class I/1 because, under normal operating conditions, all energized laser transmitting circuit packs are terminated on optical fibers which enclose the laser energy with the fiber sheath forming a protective housing. Also, a protective housing/access panel is typically installed in front of the laser circuit pack shelves The circuit packs themselves, however, may be FDA/CDRH Class I, IIIb, or IV or IEC Class 1, 1M, 3R, 3B, or 4.

Laser safety precautions for optical fiber telecommunication systems

In its normal operating mode, an optical fiber telecommunication system is totally enclosed and presents no risk of eye injury. It is a Class I/1 system under the FDA and IEC classifications.

The fiber optic cables that interconnect various components of an optical fiber telecommunication system can disconnect or break, and may expose people to laser emissions. Also, certain measures and maintenance procedures may expose the technician to emission from the semiconductor laser during installation and servicing. Unlike more familiar laser devices such as solid-state and gas lasers, the emission pattern of a semiconductor laser results in a highly divergent beam. In a divergent beam, the irradiance (power density) decreases rapidly with distance. The greater the distance, the less energy will enter the eye, and the less potential risk for eye injury. Inadvertently viewing an un-terminated fiber or damaged fiber with the unaided eye at distances greater than 5 to 6 inches normally will not cause eye injury, provided the power in the fiber is less than a few milliwatts at the near IR wavelengths and a few tens of milliwatts at the far IR wavelengths. However, damage may occur if an optical instrument such as a microscope, magnifying glass, or eye loupe is used to stare at the energized fiber end.

Laser safety precautions for enclosed systems

Under normal operating conditions, optical fiber telecommunication systems are completely enclosed; nonetheless, the following precautions shall be observed:

1. Because of the potential for eye damage, technicians should not stare into optical connectors or broken fibers
2. Under no circumstance shall laser/fiber optic operations be performed by a technician before satisfactorily completing an approved training course
3. Since viewing laser emissions directly in excess of Class I/1 limits with an optical instrument such as an eye loupe greatly increases the risk of eye damage, appropriate labels must appear in plain view, in close proximity to the optical port on the protective housing/access panel of the terminal equipment.

Laser safety precautions for unenclosed systems

During service, maintenance, or restoration, an optical fiber telecommunication system is considered unenclosed. Under these conditions, follow these practices:

1. Only authorized, trained personnel shall be permitted to do service, maintenance and restoration. Avoid exposing the eye to emissions from un-terminated, energized optical connectors at close distances. Laser modules associated with the optical ports of laser circuit packs are typically recessed, which limits the exposure distance. Optical port shutters, Automatic Power Reduction (APR), and

Automatic Power Shut Down (APSD) are engineering controls that are also used to limit emissions. However, technicians removing or replacing laser circuit packs should not stare or look directly into the optical port with optical instruments or magnifying lenses. (Normal eye wear or indirect viewing instruments such as Find-R-Scopes are not considered magnifying lenses or optical instruments.)

2. Only authorized, trained personnel shall use optical test equipment during installation or servicing since this equipment contains semiconductor lasers (Examples of optical test equipment are optical time domain reflectometers (OTDRs), Hand-held loss test sets.)
3. Under no circumstances shall any personnel scan a fiber with an optical test set without verifying that all laser sources on the fiber are turned off
4. All unauthorized personnel shall be excluded from the immediate area of the optical fiber telecommunication systems during installation and service.

Consult ANSI Z136.2, American National Standard for Safe Use of Lasers in the U.S.; or, outside the U.S., IEC-60825, Part 2 for guidance on the safe use of optical fiber optic communication in the workplace.

For the optical specifications refer to the chapter “Technical specifications” in the Product Information and Planning Guide.

Conventions used

See “Common data types” (p. 1-9) for detailed information.

Related information

The manuals related to 1675 LambdaUnite MSS are shown in the following table:

Document title	Document code
1675 LambdaUnite MSS Product Information and Planning Guide Presents a detailed overview of the system, describes its applications, gives planning requirements, engineering rules, ordering information, and technical specifications.	109671750 (365-374-195R11.0)
1675 LambdaUnite MSS User Provisioning Guide Provides step-by-step information for use in daily system operations. The manual demonstrates how to perform system provisioning, operations, and administrative tasks by use of *WaveStar*® CIT.	109671776 (365-374-196R11.0)
1675 LambdaUnite MSS Maintenance and Trouble-clearing Guide Gives detailed information on each possible alarm message. Furthermore, it provides procedures for routine maintenance, troubleshooting, diagnostics, and component replacement.	109671768 (365-374-197R11.0)
1675 LambdaUnite MSS Installation and System Turn-up Guide A step-by-step guide to system installation and set up. It also includes information needed for pre-installation site planning and post-installation acceptance testing.	109671792 (365-374-198R11.0)

Document title	Document code
1675 LambdaUnite MSS TL1 Command Guide Describes the external TL1 interface for 1675 LambdaUnite MSS in terms of TL1 command, responses, and notification definitions.	109671784 (365-374-199R11.0)
ES64U Command Line Interface Guide	109681981 (365-374-200)
ES64U SNMP Reference Guide	109681999 (365-374-201)
1675 LambdaUnite MSS Safety Guide	109510909 (365-374-159R11.0)
Documentation CD-ROM 1675 LambdaUnite MSS (all manuals on a CD-ROM)	109671800 (365-374-181R11.0)
1675 LambdaUnite MSS Software Release Description	This document is delivered with the NE software.
1675 LambdaUnite MSS Engineering and Ordering Information	Drawing ED8C948-10
1675 LambdaUnite MSS Interconnect and Circuit Information	Drawing ED8C948-20

These documents and drawings can be ordered at or downloaded from the Alcatel-Lucent Online Customer Support Site (OLCS) (https://support.lucent.com) or through your Local Customer Support.

Technical documentation

The technical documentation as required by the Conformity Assessment procedure is kept at Alcatel-Lucent location which is responsible for this product. For more information, contact your local Alcatel-Lucent representative.

Technical support

For technical support, contact your local Alcatel-Lucent customer support team. See the Alcatel-Lucent Support web site (http://www.alcatel-lucent.com/support/) for contact information.

How to comment

To comment on this information product, go to the Online Comment Form (http://www.lucent-info.com/comments/enus/) or e-mail your comments to the Comments Hotline (comments@alcatel-lucent.com).

1 General information

Overview

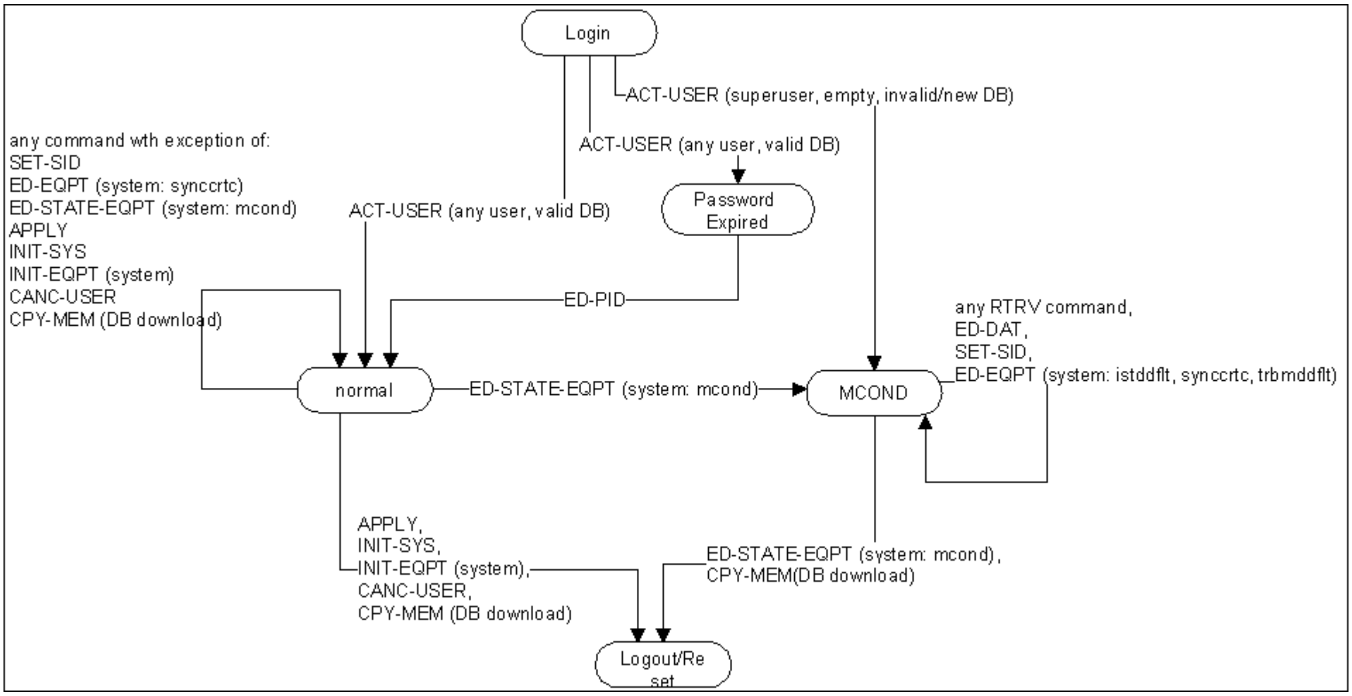
Purpose

This chapter contains introductory information.

Contents

TL1 command state diagram

The following state diagram provides an overview in which system states which TL1 commands are allowed. States and transitions are explained below the figure.



States

- *Login*: initial state, creation of user session
- *Password expired*: entered after login with valid database and when the user password has expired
- *MCOND*: system is in maintenance condition (see parameter *mcond* in RTRV-STATE-EQPT (system))
- *normal*: system is in normal mode (see parameter *mcond* in RTRV-STATE-EQPT (system)
- *Logout/Reset*: termination of a user session through logout or a command which leads to a system or CTL controller reset

Transitions

Transitions between the states are performed upon execution of the TL1 commands on the arrows between the states. An exception is the transitions from the *Login* state. This transition is automatically performed based on the database state (see parameter *dbstate* in RTRV-STATE-EQPT (system)) and the password expiration information (see parameter page in RTRV-USER-SECU).

Generic TL1 command behavior

The purpose of this chapter is to describe TL1 command behavior, which is common for all commands and therefore not stated explicitly for each single command. Deviations from the common behavior are described for each command in the detailed behavior description.

Input/output case sensitivity

For details, refer to *GR-831 section 2.7.3*.

Input syntax format

The input parser of the network element is case insensitive. The commands are recognized regardless of their capitalization. Command codes and parameter names in uppercase, lowercase, or mixed case (combinations of uppercase and lowercase within an input message, a command code, or a parameter name) are allowed.

Output syntax format

The recommended output format is for parameter names always lowercase and for value sets as defined in this document. Derivations may be acceptable.

Edit commands

ED commands modify the parameters as specified in the command. Spec block parameters are optional. Parameters which are not specified in the command remain unchanged.

It is allowed to omit all spec block parameters in the command. Such an empty command does not change any parameter value, and cause an ED notification without spec block parameters.

Data base change notifications

Object creation - ENT notification

The creation/establishment of an entity/instance is reported by means of an ENT (Enter) notification. It consists of:

- entity class name
- instance/identifier/key
- all attribute values of the instance as defined in the notification section; example: ENT-OC48 notification
- entity class name: OC48
- instance/identifier/key: port AID
- attribute values: all attribute values as defined in the corresponding ENT-OC48 notification section.

The attributes for an entity can be classified as follows:

- *input*: the reported attribute value is specified in the corresponding command request
- *modifiable*: the reported attribute value is the specified default value (with semantics of initial value) and the attribute itself is modifiable by means of corresponding ED command
- *state*: the reported attribute value is the specified default value (with semantics of initial value) and the attribute itself is modifiable directly by means of corresponding ED command. The modification can ever be triggered as side affect of a command or by an autonomous event.

Examples for the different attribute classes

- input: 1+1 line/MS protection group: attribute wkg - worker port
- modifiable: 1+1 line/MS protection group: attribute wkg - worker port
- state: port: attribute crstat - Constituent Rates Status list (autonomous change) in case of adaptive mode
- port: attribute: fsflpbkstat - Far Side Facility Loopback Status (side effect of command)

Note: Protection state information is not contained in any db change notification.

Object modification - ED notification

The modification for some attributes of an entity/instance is reported by means of an ED (Edit) notification. It consists of:

- entity class name
- instance/identifier/key

- all attribute values which were modified
  This includes all attributes which came into existence by modifying another attribute. (see example below). This also means, that attributes which are already handled internally but where not applicable up to now and therefore were not reported to the manager, must not be reported to the manager, if they became applicable. Reason: The manager cannot know the internal default values. If an ED command for an (or the) entity/instance attribute does not modify the attribute value, because (for each individual attribute) new_attribute_value = old_attribute_value, no ED notification is sent However, empty notifications, that is, notifications with no attributes and attribute values, can occur. These empty notifications do not affect the managers, but nevertheless should be avoided. Example: **`ED-OC48`** notification
- entity class name: OC48
- instance/identifier/key: port AID
- attribute values: all attribute values which have been modified.
  Thus the corresponding ED notification section describes only the superset of attributes where changes are reported. Which attributes are really contained in a concrete scenario is described either in the detailed behavior section (if the change is triggered by a command) or in autonomous event section (if the changes it triggered by a spontaneous event). Example: ED-OC48 notification of new attributes.
- entity class name: STM16
- instance/identifier/key: port AID
- action: istd is set from SDH to SONET
  This action causes that attributes come into existence by modifying *istd*. In detail this means: - attributes, which are named differently in the two standards - attributes, which are only applicable in one standard must be reported. For example, for the change from SDH to SONET: - *sdthr* (in SDH called *degthr*) - *felpm* (in SDH called *femspm*) - *z0outmode* (new parameter in SONET) must be reported.

Note: Sometimes TL1 uses a different command verb instead of “ED” for modification commands (for example, “SET”). Nevertheless, the statements of this chapter are also valid for these.

Object deletion - DLT notification

The deletion/removal of an entity/instance is reported by means of a DLT (Delete) notification. It consists of:

- entity class name
- instance/identifier/key

Example: DLT-OC48 notification

- entity class name: OC48
- instance/identifier/key: port AID.

Sequence constraints

To allow the manager a proper interpretation of the received notifications, the NE must assure same sequence constraints. This section specifies these sequence constraints.

Report object creation first

The first notification for an entity is the *db change* reporting that a new entity has been created/established. After this *establish* notification, all other types of notifications may be issued for this instance.

Report creation according to AID hierarchy

If several instances of entities are created in the context of one stimulus, the order of the *establish* notification is according to the hierarchy of the AID structure. The *establish* notification for a subordinate entity is not reported before the *establish* notification of a superior entity.

Example: The *establish* notification for a port should be sent before the *establish* notification of a tributary being a member of this port.

Report object deletion last

The last notification for an entity is the *dbchange* reporting that an entity has been deleted/removed.

Report deletion according to AID hierarchy

If several instances of entities are deleted in the context of one stimulus, the order of the *delete* notification is according to the hierarchy of the AID structure. The *delete* notification for a superior entity is not reported before the *delete* notification of a subordinate entity.

Example: The *delete* notification for a tributary should be sent before the *delete* notification of a port containing that tributary.

TL1 command validation principles

This section defines the following principles for the TL1 command validations:

- Validations to be executed
- Order of the validations
- Behavior in the case of multiple failures in a command

General validations to be executed first

On receipt of a TL1 command, the system performs a general TL1 command validation, to check:

- Correct command name
  (includes a check, whether the command is supported by the system)
- Command format
- Completeness of command parameters
- Correct spec block parameter names
  (includes a check, whether the parameter is supported by the system)
- Correct parameter syntax
  (includes a check for the maximum value range)

The error responses for the general validations are defined in the error responses section of the RTRV-HDR command.

Order of TL1 command validations

The TL1 command validations are performed in the order as given by the command parsing and processing. This requirement defines that the validations are executed in the “natural order” as the command parsing and processing steps are done.

An example would be, that the existence of an entity must be checked, before parameter values for the entity can be evaluated.

Immediate error response generation

The system generates the error response immediately after a TL1 command validation failed. This means, that the system does not store the results of several validations and later decide, which error response has to be generated.

This behavior is important, in the case of multiple failure conditions in the command. The error response which will be returned depends on the first failed validation during command processing.

Common data types

The following data types are used in the description of TL1 messages, the definition is given in Backus-Naur Form. Characters of the Backus-Naur Form are enclosed in “ when they are meant as literal in the definition.

<table>
<tr><th>GR-831-CORE Definition</th><th>Name used in TL1 messages</th><th>Definition</th></tr>
<tr><td>nil</td><td>nil</td><td>nil, symbolizes the absence of any other thing</td></tr>
<tr><td>let</td><td>letter</td><td>A | B | … | Z | a | b | … | z</td></tr>
<tr><td>dig</td><td>digit</td><td>0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9</td></tr>
<tr><td>ident</td><td>identifier</td><td><letter> (<letter> | <digit>) *</td></tr>
<tr><td>sym name</td><td>symbolic name</td><td><digit>* ((<letter> | ″+″ | # | %) <digit>*)+</td></tr>
<tr><td>alphanum</td><td>alphanumeric string</td><td>(<letter> | <digit>) +</td></tr>
<tr><td>dec num</td><td>decimal number</td><td>[D″]<digit>* [.] <digit>+</td></tr>
<tr><td>arith exp</td><td>arithmetic expression</td><td>″(″ [″+″ | –] <digit>*[.]<digit>+ ″)″</td></tr>
<tr><td>hex num</td><td>hexadecimal number</td><td>[H″] (<digit> | A | B | C | D | E | F)+</td></tr>
<tr><td>oct num</td><td>octal number</td><td>[O″] (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7)+</td></tr>
<tr><td>bin num</td><td>binary number</td><td>[B″] (0 | 1)+</td></tr>
<tr><td>keyed num</td><td>keyed number</td><td>[K″] (<digit> | A | B | C | D | ″*″ | #)+</td></tr>
<tr><td>num str</td><td>numeric string</td><td><digit>+</td></tr>
<tr><td>integer</td><td>integer</td><td><numeric string> | ″(″ [″+″ | -] <digit>+ ″)″</td></tr>
<tr><td>text str</td><td>text string</td><td>″(\″ | \\ | <any character except ″ or \>)* ″</td></tr>
<tr><td>inner str</td><td>inner string</td><td>\″ (″″ | \\ | <any character except ″ or \>)* \″</td></tr>
<tr><td>aid</td><td>AID</td><td>see Chapter 2, “AID structure”.</td></tr>
<tr><td></td><td>AID list</td><td><AID> (&<AID>)*</td></tr>
<tr><td>tid</td><td>TID</td><td><identifier>(-<identifier>) *|<text string> see also RTRV-HDR for 1675 LambdaUnite MSS specific constraints</td></tr>
<tr><td></td><td>character string</td><td>any character except: ; , : ″</td></tr>
<tr><td></td><td>value set</td><td>set of predefined <character strings></td></tr>
<tr><td></td><td>alphanumeric value set</td><td>set of predefined <alphanumeric strings></td></tr>
<tr><td></td><td>numeric value set</td><td>set of predefined <numeric strings></td></tr>
<tr><td></td><td>AID string</td><td>(letter | digit | _ | - | & | #)+</td></tr>
</table>

Validation of character set restrictions

If any parameter of the type *Text String*, *Inner String*, or *Character String* has a restricted character set, then the system validatse the parameter that only the allowed character set is used. The character set validation ensures that:

- The system function is not impaired
- The interworking with systems of the same type (1675 LambdaUnite MSS) is not disturbed.

The detailed character set restrictions are defined either in the parameter range definition or in specific validation rules of the TL1 commands.

The system does not consider character set restrictions from other NE types, which are intended for interworking, and from management systems or CIT. Restricting the character sets to meet the need of all possible interworking partners would lead to a very limited character set. This would be an unnecessary restriction for users, who usually have only some of the possible NE types in the network and only one type of management system.

It is assumed that management systems, *WaveStar*® CIT and other NE types use the same principles for character set validation, that is, perform validations to ensure their own system function.

Recommended handling of character set restrictions

To avoid network operation impairments caused by violations of character set restrictions, the user shall

- collect the allowed character sets for the provisionable parameters from all NE types in the network and from the management system
- define suitable character sets which meet the restrictions of all NE types.

It is the users task to ensure that the defined character sets are properly used (for example, through trainings within the organization).

IP addresses

During provisioning, IP addresses are not recognized as different in case they only differ in leading zeroes.

In case IP addresses are configured using leading zeroes in any of their fields, during provisioning the system does not recognize IP addresses as equal that differ only by such leading zeroes. When using IP addresses for communication, however, leading zeroes are ignored. As a consequence, it would then be possible, that the same IP address be used in two applications that need different ones, for example L1 RC IP address and ENNI SC PC IP address.

When provisioning IP addresses, do not use leading zeroes.

Definitions

Definition of the directions used for TransMUX

The TransMUX application has two sides:

- Asynchronous side
- Synchronous side

On each side, the egress (outgoing) direction is defined in a way that it points to the other TransMUX application side.

That means:

- The egress (outgoing) direction of the asynchronous side directs to the synchronous side.
- The egress (outgoing) direction of the synchronous side directs to the asynchronous side.

For the ingress (ingoing) direction, the opposite applies.

2 AID structure

Overview

Purpose

This section contains the requirements for Access Identifiers (AID) for the TL1 interface of the 1675 LambdaUnite MSS.

Contents

Introduction

Overview

Purpose

This section defines the names which will be used to identify physical and logical entities for input and output at the TL1 Interface for the 1675 LambdaUnite MSS products.

This document re-uses the general terminology and BWM Access Identifier (AID) Syntax. It is assumed that the reader is familiar with the terms and syntax.

Contents

AID syntax

General

The following specification characters will be used throughout this document as a vehicle for defining the syntax:

<table>
<tr><th>Rule</th><th>Description</th></tr>
<tr><td>::=</td><td>Assignment of a lexical rule to a token or token class.</td></tr>
<tr><td>literal</td><td>Characters which must appear literally as typed appear in typewrite font.</td></tr>
<tr><td>|</td><td>The “exclusive or” token means that exactly one item in the list must be present.</td></tr>
<tr><td>{}</td><td>Items in {} mean that one of the items in the set must be present.</td></tr>
<tr><td>[]</td><td>Items in [] mean that one of the items in the set may optionally be present.</td></tr>
<tr><td>()</td><td>Items in () can be used for grouping to define different precedence for the operators |, ^{n}, and *.</td></tr>
<tr><td>-</td><td>A dash indicates a range of values where any item in the range may be present.</td></tr>
<tr><td>,</td><td>A “,” is used as a list item separator in the specification grammar.</td></tr>
<tr><td>a^*</td><td>A superscript “*” specifies 0 or more repetitions of the previous token that can be present.</td></tr>
<tr><td>a^+</td><td>A superscript “+” specifies 1 or more repetitions of the previous token that can be present.</td></tr>
<tr><td>a^n</td><td>A superscript “n” is a number which specifies the number of repetitions of the previous token that can be present.</td></tr>
<tr><td>“”</td><td>Blanks between tokens in an expression means to concatenate the tokens.</td></tr>
</table>

Access Identifier (AID) syntax

AIDs can be either Simple_AIDs, or Compound_AIDs. AID definitions will use the following syntax:

let ::= {**A-Z**,**a-z**,_}

dig ::= {**0-9**}

AID_field ::= (let (let | dig)*)

Simple_AID ::= (AID_field [Simple_AID]) | **#**

Compound_AID ::= Simple_AID(–Simple_AID)*

The character "–" (dash) will be used as the Simple_AID separator in compound AIDs.

AID_List::=AID_unit(, AID_unit)*

AID_unit::=Simple_AID | Compound_AID

The character "," (comma) will be used as the Simple_AID, Compound_AID separator in AID_Lists.

If a Simple_AID is <null>, the "**#**" character is used to specify a value of <null> for an AID_field.

Groups of entities of the same type can be specified by using the value **all** or *someprefix* **all** in a Simple_AID.

The legal places to use this value are specified in the appropriate commands below. The convention used in the command descriptions is that the value **all** is not allowed unless explicitly stated otherwise.

In general **all** is only allowed for input AIDs, that means, when used in a TL1 request.

For the following requirements see also the definition of compound AIDs below.

The value **all** on bay and shelf level is only allowed for **RTRV-EQPT** on bay and shelf level. It is not allowed on this positions for entities further down in the hierarchy like slot, port, tributary.

Example 1

valid: **all–all**

invalid: **all–all #–#–1–3–1**

The value **all** on the AID field for slot, CP, port or tributary is only allowed

- when it is the last AID field or
- when there is already an **all** on the AID field following the AID field.

Example 2

valid: `1–1–#–#–1–3–all`

valid: `1–1–#–#–1–all`

valid: `1–1–#–#–all–cp`

valid: `1–1–#–#–1–all–all`

invalid: `1–1–#–#–1–all–1`

Examples for lower-order tributaries:

valid: `1–1–#–#–1–3–1–all` (all LO tributaries for one path termination)

valid: `1–1–#–#–all–all–all–all` (all lower-order tributaries)

valid: `1–1–#–#–all–all–all–1`

System level AIDs

Overview

Purpose

This section contains all system level AIDs.

Contents

System AID

To identify the entire system the following AID is used:

SystemAID ::= **`system`**

MDI/MDO

General

The system has miscellaneous discrete outputs and inputs which are named without any hierarchical, positional information in the AID.

The system provides 8 input and 8 output miscellaneous discrete alarm points which can be assigned user-friendly names so that when the user wants to assert an output point, or receives an alarm from one of the input points, a name is used which reflects what the alarm point is physically connected to.

AIDs for MDIs, MDOs

The following AIDs are used to identify the MDIs/MDOs:

Miscellaneous_in_discreteAID ::= **`misc_in`**{**`1-8`**} | **`misc_inall`**
Miscellaneous_out_discreteAID ::= **`misc_out`**{**`1-8`**} | **`misc_outall`**

Timing

AIDs for external timing references

The timing references can come from two external DS1 or E1 (2 MHz or 2 Mbit/s) or 64 kbit/s (future) signals:

External_timing_ref_AID ::= **extref1** | **extref2** | **extrefall** | **NONE**

The value NONE indicates that no external timing reference is available.

In the system, maximal 32 I/O slots are present. Every I/O slot can supply one timing reference signal extracted from the incoming data. Out of those six can be used as line timing input reference.

AIDs for line timing references

The AIDs which can be used for the input selection are:

Line_timing_ref_AID ::= **line**{**1-6**} | **lineall** | **NONE**

The value NONE indicates that no line timing reference is available.

AIDs for timing input ports

The system has two timing input ports. The AIDs for these timing input ports are:

External_Timing_input_port_AID ::= **exttmg0** | **exttmg1** | **exttmgall**

AIDs for timing output ports

The system has two timing output ports. The AIDs for these timing output ports are:

External_Timing_output_port_AID ::= **exttmg0_out** | **exttmg1_out** | **exttmg_outall**

AID for timing output

The AID for timing output is

Timing_Output_AID ::= **outtmg**

For provisioning or retrieving information related to the system clock the SystemAID is to be used.

OSI nodes

AIDs for OSI nodes

The system can support up to 8 OSI nodes, which are used to allow that the system is part in several routing domains.

The AIDs to access the OSI nodes are:

OSI_node_AID ::= **`osinode`**{**`1-8`**} | **`osinodeall`**

SCN node

AID for SCN node

The system shall support one SCN node, which is used to implement the IP stack for SCN communication related to ONNS.

The AID to access the SCN node is:

SCN_node_AID ::= **scnnode**

UNI node

AIDs for UNI node

The system supports one UNI node, which is used to implement the IP stack for UNI communication.

The AID to access the UNI node is:

UNI_node_AID ::= uninode

ENNI node

AID for ENNI Node

The system shall support one ENNI node, which is used to implement the IP stack for ENNI communication.

The AID to access the ENNI node is:

ENNI_node_AID ::= **`OSI_node_AID | SCN_node_AID`**

General node

General Node AID

The system shall support a general node AID, which can be used to address any of the particular node AID types of above.

The general node AID is defined as:

Node_AID ::= **`OSI_node_AID | SCN_node_AID`**

ASTN node

ASTN Node AID

The system shall support a ASTN node AID, which can be used to address a ONNS node.

The ASTN node AID is defined as:

ASTN_Node_AID ::= last 4 bytes of the backpl, seperated by “–” (dash)

Link

The AIDs defined in this chapter are used to identify entities, which are defined by a relationship between two objects. E.g. a communication link, which is defined by the two end points, or a neighborship, which is defined by the two neighboring objects.

Link AID

To identify the entire communication link for ONNS between two nodes the following AID is used:

- Initiator_ASTN_Node_AID ::= ASTN_Node_AID
- Responder_Node_AID ::= ASTN_Node_AID
- LinkAID ::= Initiator_ASTN_Node_AID, Responder_ASTN_Node_AID

RC Neighbor AID

To identify the neighborship between a Routing Controller and a remote Routing Area, the following AID shall be used:

- Neighbor_RA_AID ::=
  4 byte values in the range of 0…255 each, separated by "-" (dash). The values shall be derived from the Remote Routing Area ID.
- RC_Neighbor_AID ::= ASTN_Node_AID,Neighbor_RA_AID
  The RC Neighbor AID identifies an E-NNI link by specifying the ASTN Nnode ID of the node to which the link is attached and the configured peer RA ID.

UNI LAN Control Channel Link AID

To identify the entire UNI control channel link between UNI-N and UNI-C via LAN port, the following AID shall be used:

- UNI_ControlChannel_Destination_AID ::=
  4 byte values in the range of 0…255 each, separated by"-" (dash). The values shall be derived from the destination UNI Node ID.
- UNILANCCLinkAID ::= LanPortAID,UNI_ControlChannel_Destination_AID.

ENNI LAN Control Channel Link AID

To identify the entire ENNI control channel link between ENNI Signaling Protocol controllers via LAN, the following AID shall be used:

- ENNI_Neighbor_Signalling_PC_ID ::=
  4 byte values in the range of 0…255 each, separated by"-" (dash). The values shall be derived from the destination ENNI Node ID.
- ENNILANCCLinkAID ::= LanPortAID,ENNI_Neighbor_Signalling_PC_ID

Hierarchically specified AIDs

Overview

Purpose

Below the system level AIDs, other entities are specified in a hierarchical structure.

Contents

Compound AID syntax overview

The AID structure defined for a compound AID is shown in the next table. For each AID field further information can be found in the subsequent subsections. Examples of compound AIDs can be found in "AID examples" (p. 2-62).

Entity	AID Field						
	Bay/ Rack	**Shelf/ Subrack**	**Prot Grp**	**Side Line**	**Slot**	**CP/ SPort/ NVM**	**STS-1/ AU-3/ VCG Trib**
Bay/Rack	1						
Shelf/ Subrack	1	1					
Protection Group	1	1	Simple_ ProtectionGroupAID				
Fan Unit	1	1	Simple_FanAID				
PI	1	1	Simple_PowerAID				
TI	1	1	Simple_ TimingInterfaceAID				
Backplane	1	1	Simple_BackplaneAID				
User Panel	1	1	Simple_UserPanelAID				
CI-CTL	1	1	Simple_CiCtlAID				
User Byte Ports	1	1	Simple_ UserBytePortAID				
LAN Ports	1	1	Simple_LanPortAID				
Slot	1	1	#	#	Simple_ SlotAID		
CP	1	1	#	#	Simple_ SlotAID	**cp**	
NVM	1	1	#	#	{**ctlw**| **ctlp**}	**nvm**	
SPort	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	
STS-1/ AU-3 Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *
STS-3c/ AU-4 Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *
STS-12c/ AU-4-4c Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *
STS-48c/ AU-4-16c Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *

Entity	AID Field						
	Bay/ Rack	Shelf/ Subrack	Prot Grp	Side Line	Slot	CP/ SPort/ NVM	STS-1/ AU-3/ VCG Trib
STS-192c/ AU-4-64c Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *
STS-768c/ AU-4-256c Trib	1	1	#	#	Simple_ SlotAID	Simple_ SPortAID	Simple_STS1_ NumAID *
* See table below							

The STS-3c/AU-4, STS-12c/AU-4-4c, STS-48c/AU-4-16c, STS-192c/AU4-64c, STS-768c/AU4-256c tributaries also use the STS-1/AU-3 AID field. They are identified by the first STS-1/AU-3 number of the signal as indicated by the table below:

Signal	MS2.5G/S3[-x] Adaptation (x in {3c, 12c, 48c})	MS10G/S3[-x] Adaptation (x in {3c, 12c, 48c, 192c})	MS40G/S3[-x] Adaptation (x in {3c, 12c, 48c, 192c, 768c})
S3 (STS-1/AU-3)	1, 2, 3, … , 48	1, 2, 3, … , 192	1, 2, 3, … , 768
S3-3c (STS-3c/AU-4)	1, 4, 7, … , 46	1, 4, 7, … , 190	1, 4, 7, … , 766
S3-12c (STS-12c/AU-4-4c)	1, 13, 25, 37	1, 13, 25, … , 181	1, 13, 25, … , 757
S3-48c (STS-48c/AU-4-16c)	1	1, 49, 97, 145	1, 49, 97, … , 721
S3-192c (STS-192c/AU-4-64c)	n/a	1	1, 193, 385, 577
S3-768c (STS-768c/AU-4-256c)	n/a	n/a	1

AID field lengths

The following table lists the AID field length specifications. The maximum length of an AID, including the separating “–”, must not exceed 23 characters:

Field	AID definitions	Length (char)
Bay (Rack)	BayID	1
Shelf (Subrack)	Simple_ShelfID	1
Slot	Simple_SlotAID	5
CP	Simple_CpAID | Simple_NvmAID	3
Socket	Simple_SocketAID	5
Optical Module	Simple_OmAID	3
Port	Simple_SPortAID | Simple_EPortAID | Simple_VCGAID	2
OTN Port	Simple_OTNPortAID	3

Field	AID definitions	Length (char)
TransMUX Port	Simple_TMuxAPortAID| Simple_TMuxSPortAID	5
Trib	Simple_STS1NumAID | Simple_VCGNumAID	3
HO Path Termination	Simple_PTNumAID	3
HO PTF TransMUX	Simple_TMuxAPTNumAID| Simple_TMuxSPTNumAID	1
LO Trib	Simple_LONumAID	2
LO Path Termination	Simple_TMuxSLONumAID	2
DS1 NIM	Simple_DS1NumAID	3

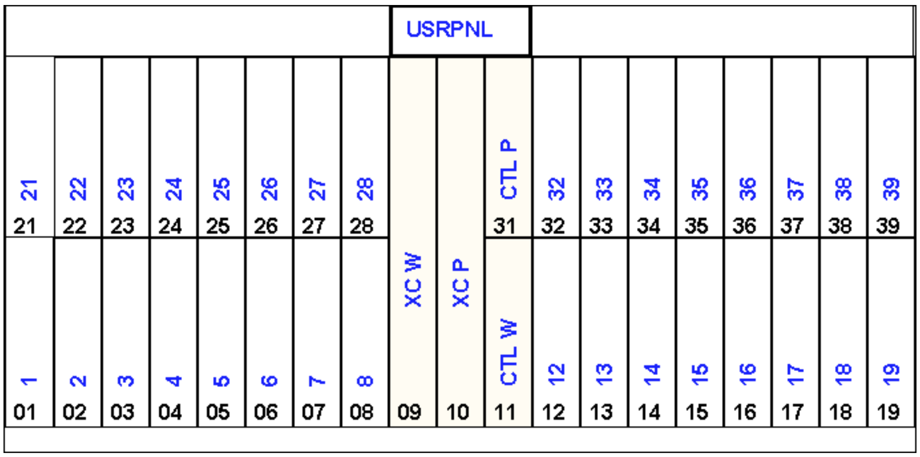
Range specifications

Various TL1 commands allow for the specification of a range of entities as input parameters. These ranges are specified in the AID definitions in this document which allow the use of the term all in one or more AID Fields within a Simple_AID. Details of where these values are allowed is detailed in the AID parameter description per command.

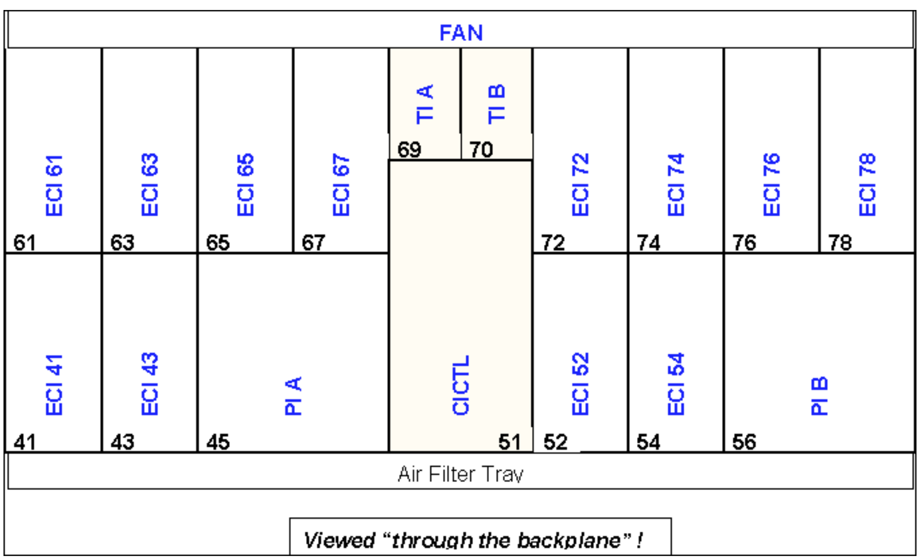
Physical layout

Frontside

The following graphics show the physical layout of the dual unit row (DUR) subrack (frontside) and the naming conventions:



Rearside



Bay (rack) and shelf (subrack)

The 1675 LambdaUnite MSS system uses a single unit row (SUR) subrack in the REG40 configuration or a dual unit row (DUR) subrack in the LXC40 configuration.

Up to two DUR, three SUR or combinations 2 SUR/1DUR, 1SUR/1DUR fit into one rack (bay). From the AID definition this relationship will not become visible.

AID for bay (rack)

The BayAID is used to identify the rack (bay) of the system:

BayAID ::= **1** | **all**

AID for shelf (subrack)

The ShelfAID is used to identify the subrack type (shelf):

Simple_ShelfAID ::= **1** | **all**

ShelfAID ::= BayAID–Simple_ShelfAID

Note: Different configurations will be distinguished by a the parameter *shlftp* in RTRV-EQPT for subrack (shelf).

Fans

AIDs for fan

There is one fan unit in the system. The FanAID is used to identify the fan unit in the subrack:

Simple_FanAID ::= **`fan`** | **`fanall`**

FanAID ::= BayAID–Simple_ShelfAID–Simple_FanID

Power interface

AIDs for power interfaces

There are two power interface units PI(A) and PI(B) in the subrack, which are located at the rear side. The PowerAID is used to identify the power interface in the subrack:

Simple_PowerAID ::= **`pia`** | **`pib`** | **`piall`**

PowerAID ::= BayAID–Simple_ShelfAID–Simple_PowerAID

Backplane

AID for backplane

The subrack has a backplane with an inventory EEPROM. The BackplaneAID is used to identify the backplane in the subrack:

Simple_BackplaneAID ::= **`backpl`**

BackplaneAID ::= BayAID–Simple_ShelfAID–Simple_BackplaneAID

User panel

AID for user panel

There is one user panel in the subrack. The UserPanelAID is used to identify the user panel in the system:

Simple_UserPanelAID ::= **`usrpnl`**

UserPanelAID ::= BayAID–Simple_ShelfAID–Simple_UserPanelAID

CI-CTL

AID for CI-CTL

The system has a connection interface with e.g. LAN and user byte ports (CI-CTL). The CiCtlAID is used to identify the CI-CTL in the subrack:

Simple_CiCtlAID ::= **`cictl`**

CiCtlAID ::= BayAID–Simple_ShelfAID–Simple_CiCtlAID

User byte ports

AIDs for user byte ports

The system has six ports to access user bytes which are located on the CI-CTL. The UserBytePortAID is used to identify a user byte port in the subrack:

Simple_UserBytePortAID ::= **`ubio1`** | **`ubio2`** | **`ubio3`** | **`ubio4`** | **`ubio5`** | **`ubio6`** | **`ubioall`**

UserBytePortAID ::= BayAID–Simple_ShelfAID–Simple_UserBytePortAID

LAN ports

AIDs for LAN ports

The system has four LAN ports which are located on the UPL (LAN1) and CI-CTL (LAN2, LAN3). A LAN4 is currently only HW prepared and has no further functionality. The LanPortAID is used to identify a LAN port in the subrack:

Simple_LanPortAID ::= **`lan1`** | **`lan2`** | **`lan3`** | **`lanall`**

LanPortAID ::= BayAID–Simple_ShelfAID–Simple_LanPortAID

Timing interfaces (TI)

AIDs for timing interfaces

The system has up to two timing interfaces TI(A) and TI(B) located on the rear side of the subrack. The TimingInterfaceAID is used to identify a timing interface in the subrack:

Simple_TimingInterfaceAID ::= **`tia`** | **`tib`** | **`tiall`**

TimingInterfaceAID ::= BayAID–Simple_ShelfAID–Simple_TimingInterfaceAID

Electrical Connector Interface (ECI)

The system has up to 12 electrical connector interfaces (ECI) located on the rear side of the subrack. Note: Only the upper shelf row can be used for EP155 and EP51 configurations.

AIDs for electrical connector interfaces

The EciAID shall be used to identify a electrical connector interface in the subrack:

Simple_EciAID::=
`eci61 |eci63 | eci65 |eci67 | eci72 | eci74 | eci76 | eci78 | eciall`

EciAID::=BayAID–Simple_ShelfAID–Simple_EciAID

Legal Simple_EciAIDs depend on the type of ECI as shown in the following table:

ECI type	Legal ECI AIDs
ECI (for EP155)	eci61 | eci63 | eci65 | eci67 | eci72 | eci74 | eci76 | eci78
ECI51 (for EP51)	eci61 | eci65 | eci72 | eci76

Slot

The 1675 LambdaUnite MSS in a DUR subrack has in total two rows of circuit packs as shown in figure in “Frontside” (p. 2-25). The slots for the cross connect pack (XC) and for the controller pack (CTL) are distinguished from the universal slots (US). The postfix P(rotection) or W(orking) for the XC and CTL indicates its role in the 1+1 equipment protection relationship.

AIDs for universal slots

The UnivSlotAID is used to identify a universal slot in the system:

Simple_UnivSlotAID ::= {**1 … 8**, **12 … 19**, **21 … 28**, **32 … 39**} | **usall** | **all**

UnivSlotAID ::= BayAID–Simple_ShelfAID–#–#–Simple_UnivSlotAID

Note: The us-prefix and leading zeroes have to be removed from Simple_UnivSlotAID. This change is not reflected in all examples below.

Note: The value “**all**” was added as an additional option to iterate over all slots not considering the type of slot.

AIDs for cross-connect slots

The XCSlotAID is used to identify a slot useable for an XC circuit pack in the system:

Simple_XCSlotAID ::= **xcp** | **xcw** | **xcall** | **all**

XCSlotAID ::= BayAID–Simple_ShelfAID–#–#–Simple_XCSlotAID

Note: The value “**all**” was added as an additional option to iterate over all slots not considering the type of slot.

AIDs for controller slots

The CTLSlotAID is used to identify a slot useable for a CTL circuit pack in the system:

Simple_CTLSlotAID ::= **ctlp** | **ctlw** | **ctlall** | **all**

CTLSlotAID ::= BayAID–Simple_ShelfAID–#–#–Simple_CTLSlotAID

Note: The value “**all**” was added as an additional option to iterate over all slots not considering the type of slot.

AIDs for slots

The SlotAID is used to identify a slot in the system:

Simple_SlotAID ::= Simple_UnivSlotAID | Simple_XCSlotAID | Simple_CTLSlotAID | **all**

SlotAID ::= BayAID–Simple_ShelfAID–#–#–Simple_SlotAID

The “#” in the SlotAID were used in the CG to identify protection group information for ports and tributaries. This addressing scheme is not supported.

OP40G slot restrictions

Since the OP40 is four slot wide, only the following Simple_UnivSlotAIDs are allowed:

Simple_SlotAIDOP40 ::= {**`4`**, **`8`**, **`15`**, **`19`**, **`24`**, **`28`**, **`35`**, **`39`**}

SlotAIDOP40 ::= BayAID–Simple_ShelfAID–#–#–Simple_SlotAIDOP40

Circuit pack

AIDs for circuit packs

cp can be used in the port field to identify a circuit pack:

Simple_CpAID ::= **cp**

CpAID ::= SlotAID–**cp**

The AID for circuit packs does not include the LOXC packs.

AIDs for LOXC packs

loxc can be used in the port field to identify a LOXC pack:

Simple_LoxcAID ::= **loxc**

LoxcAID ::= SlotAID–**-loxc**

The special LOXC circuit pack AID is needed to distinguish the LOXC pack from the I/O packs in the universal slots.

AIDs for all circuit packs

cp can be used in the port field to identify a circuit pack independent of the circuit pack type:

Simple_CpAID ::= **cpall**

CpAllAID ::= SlotAID–**-cpall**

The AID for all circuit packs includes the LOXC packs.

Socket

Circuit packs may contain sockets where pluggable interfaces can be plugged in.

AIDs for sockets

The Simple_SockedAID shall be used in the port field to identify a socket contained on a circuit pack which is inserted in a slot.

Simple_SocketAID ::=
sc1 | sc2 | sc3 | sc4 | sc5 | sc6 | sc7 | sc8 | sc9 | scall

SocketAID ::=BayAID-Simple_ShelfAID-#-#-Simple_UnivSlotAID-Simple_SocketAID

Legal Simple_SocketAIDs depend of the CP provisioned for the slot as shown in the following table:

Circuit Pack Type Functional Name / Qualifier	Legal Socket AIDs	Valid in Release
OPLB/PAR8	sc1 | sc2 | sc3 | sc4 | sc5 | sc6 | sc7 | sc8 | scall	4.1 and later
OP2G5D/PAR8	sc1 | sc2 | sc3 | sc4 | sc5 | sc6 | sc7 | sc8 | scall	4 and later
OP10D/PAR2	sc1 | sc2 | scall	4.1 and later
OPT2G5/PAR3	sc1 | sc2 | sc3 | scall	5 and later
OP2G5/PAR4	sc1 | sc2 | sc3 | sc4 | scall	7 and later
GE10PL1/1A8	sc1 | sc2 | sc3 | sc4 | sc5 | sc6 | sc7 | sc8 | sc9 | scall	8 and later

The system supports only a one to one relation between sockets and ports. Example: port no. 1 (1-1-#-#-1-1) is related to a socket no.1 (1-1-#-#-1-sc1) .

SDH/SONET port

AIDs for SDH/SONET ports

The SPortAID is used to identify SDH/SONET ports in the system:

Simple_SPortAID ::= {1 - 36 | all}

SPortAID ::= UnivSlotAID–Simple_SPortAID

Legal Simple_SPortAIDs depend on the type of CP provisioned for the slot as shown in the following table:

Functional Name	Legal Port AIDs	Restrictions
EP155	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**} | **all**	none - valid for all supported qualifiers
OP155	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**, **9**, **10**, **11**, **12**, **13**, **14**, **15**, **16**} | **all**	none - valid for all supported qualifiers
OP622	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**, **9**, **10**, **11**, **12**, **13**, **14**, **15**, **16**} | **all**	none - valid for all supported qualifiers
OPLB	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**} | **all**	none - valid for all supported qualifiers
OP2G5	{**1**, **3**} | **all**	qualifier = PARENT
OP2G5	{**1**, **2**, **3**, **4**} | **all**	qualifier != PARENT
OP2G5D	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**} | **all**	none - valid for all supported qualifiers
OPT2G5	{**1**, **2,3**} | **all**	none - valid for all supported qualifiers
OP10	**1** | **all**	none - valid for all supported qualifiers
OP10D	{**1**, **2**} | **all**	none - valid for all supported qualifiers
OP40	**1** | **all**	none - valid for all supported qualifiers
EP51	{**1 - 36** } | **all**	none - valid for all supported qualifiers

The table above does not take into account the release information when a certain pack is supported.

The number of legal port AIDs shown in the table above does not necessarily reflect the capabilities of the actual circuit pack w.r.t. port numbers. Additional rows are added for the packs if they differ in terms of port numbers. The functional qualifier is then used as identification criteria - see restriction column.

The definition and range of *SPortAID* for OP10 also applies to a 10GE_WANPHY port variant of a OP10 pack due to reasons of simplicity.

OTN ports

AIDs for OTN ports

The OTNPortAID is used to identify OTN ports in the system: Simple_OTNPortAID ::= ot1 | otall

OTNPortAID ::= UnivSlotAID–Simple_OTNPortAID

Legal Simple_OTNPortAIDs depend on the type of CP provisioned for the slot as shown in the following table:

Circuit Pack Type	Legal Port AIDs	Valid for
OP10/XTTL OP10/XTTC	ot1 | otall	R8.0 and later

Adjunct ports

Adjunct (virtual) ports

In order to be able to address the entities of the TransMUX pack:

- 48 adjunct ports are introduced on the *asynchronous* side
- 48 adjunct ports are introduced on the *synchronous* side

AIDs for TransMUX asynchronous adjunct ports

The TMuxAPortAID shall be used to identify the TransMUX asynchronous adjunct ports in the system:

Simple_TMuxAPortAID ::= tma{1 - 48} | all

TMuxAPortAID ::= UnivSlotAID–Simple_TMuxAPortAID

AIDs for TransMUX synchronous adjunct ports

The TMuxSPortAID shall be used to identify the TransMUX synchronous adjunct ports in the system:

Simple_TMuxSPortAID ::= tma{1 - 48} | all

TMuxSPortAID ::= UnivSlotAID–Simple_TMuxSPortAID

Asynchronous ports

AIDs for asynchronous ports

The APortAID is used to identify asynchronous ports in the system: Simple_APortAID ::= { 1 - 36 }| all

APortAID ::= UnivSlotAID–Simple_APortAID

Legal Simple_APortAIDs depend on the type of CP provisioned for the slot as shown in the following table:

Circuit Pack Type	Legal Port AIDs
EP51	{**1 - 36**} | **all**

Circuit Pack Type	Ethernet Port Rate	Legal Port AIDs	Valid in Release
GE1	1 GbE	{1, 2, 3, 4} | all	2 and later
GE10PL1/1A8	1 GbE	{1, 2, 3, 4, 5, 6, 7, 8} | all	8 and later
GE10PL1/1A8	10 GbE	9 | all	8 and later

GbE port

AIDs for Gigabit Ethernet ports

The EPortAID can be used to identify 1 Gigabit and 10 Gigabit Ethernet LANports in the system:

Simple_EPortAID ::= **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **all**

EPortAID ::= UnivSlotAID–Simple_EPortAID

Legal Simple_EPortAIDs depend on the type of CP provisioned for the slot as shown in the following table:

Circuit Pack Type	Ethernet Port Rate	Legal Port AIDs	Valid in release
GE1	1 GbE	{**1**, **2**, **3**, **4**} | **all**	2 and later
GE10PL1/1A8	1 GbE	{**1**, **2**, **3**, **4**, **5**, **6**, **7**, **8**} | **all**	8 and later
GE10PL1/1A8	10 GbE	**9** | **all**	8 and later

Virtual concatenation group (VCG)

Virtual concatenation groups (VCG) represent the interface between the Ethernet Switch and the Ethernet over SDH/SONET function. Sometimes they are also denoted as WAN ports.

AIDs for VCG

The VCGAID can be used to identify VCG ports in the system:

VCGGrpID ::= { 1, …, 126 } ∪ { 129, …, 254 }

Simple_VCGAID ::= **vvVCGGrpID** | **all**

VCGAID ::= UnivSlotAID–Simple_VCGAID

Legal VCG AIDs depend on the type of CP provisioned for the slot as shown in the following table:

Circuit Pack Type	Port Rate	Legal VCG Port AIDs	Valid in release
GE1	1 GbE	{**v1**, **v2**, **v3**, **v4**} | **vall**	2 and later
OPT2G5	STM16/OC48	{**v1**, **v2**, **v3**} | **vall**	5 and later
GE10PL1/1A8	1 GbE	{**v1**, **v2**, **v3**, **v4**, **v5**, **v6**, **v7**, **v8**} | **vall**	8 and later
GE10PL1/1A8	10 GbE	**v9** | **vall**	8 ans later

AU-3/STS-1 tributary

AIDs for AU-3/STS-1 tributaries

The STS1NumAID can be used to identify SDH/SONET tributaries in the system:

Simple_STS1NumAID ::= { **1**, **2**, … , **768** } | **all**

STS1NumAID ::= SPortAID–Simple_STS1NumAID

Legal Simple_STS1NumAIDs (STS-1/AU-3 numbers) depend on the type of CP provisioned for the slot and may depend on the provisioned pluggable interface as shown in the following table:

Circuit Pack Type	Legal STS-1/AU-3 Number	Restriction
EP51	1	none
EP155	{1 … 3}	none
OP155	{1 … 3}	none
OP622	{1 … 12}	none
OPLB	{1 … 3}	Pluggable Interface = OM155
OPLB	{1 … 12}	Pluggable Interface = OM622
OP2G5	{1 … 48}	none
OP2G5D	{1 … 48}	none
OP10	{1 … 192}	none
OP10D	{1 … 192}	none
OP40	{1 … 768}	none

Virtual concatenation group (VCG) tributary

AIDs for VCG tributaries

The VCGNumAID can be used to identify VCG tributaries in the system:

Simple_VCGNumAID ::= { **1**, … , **192** } | **all**

VCGNumAID ::= VCGAID–Simple_VCGNumAID

VCG tributaries denote a group of X virtually concatenated STSs/VCs which depends on the type of CP provisioned for the slot and the VCG mapping.

- GE1: STS-1-Xv (X ε {1, …, 21} or VC-4-Xv (X ε{1, …, 7})
- OPT2G5: STS-3c-17v/VC-4-17v
- GE10PL1 (1 GbE Port): STS-1-Xv (X = {1 … 21} or STS-3c-Xv/VC-4-Xv (X={1 … 7})
- GE10PL1 (10 GbE Port): STS-1-Xv (X = {1 … 192} or STS-3c-Xv/VC-4-Xv (X={1 … 64})
- ES64U (VCG): STS-1-Xv (X = {1 … 192}

Legal Simple_VCGNumAIDs (VCG numbers) depend on the type of CP provisioned for the slot and the VCG type (mapping) as shown in the following table:

Circuit Pack Type / VCG mapping	Legal VCG Number
GE1/STS1-VC3	{1 … 21}
GE1/STS3c-VC4	{1, 4, 7, 10, 13, 16, 19}
OPT2G5/STS3c-VC4	{1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49}
GE10PL1 (1 GbE port)	{1 … 21}
GE10PL1 (10 GbE port)	{1 … 192}

Higher-order path terminations

AIDs for higher-order path terminations

The PTNumAID shall be used to identify higher-order path terminations in the system:

Simple_PTNumAID ::= { **1**, **2**, …, **768** } | **all**

PTNumAID ::= SPortAID-Simple_PTNumAID

Legal Simple_PTNumAIDs depend on the type of CP provisioned for the slot and may depend on the provisioned Pluggable Interface as shown in the following table:

Circuit Pack Type / VCG mapping	Legal PT Number	Restrictions
EP51	1	For asynchronous ports only
EP155	1	none
OP155	{1 … 3}	none
OP622	{1 … 12}	none
OPLB	{1 … 3}	Pluggable Interface = OM155
OPLB	{1 … 12}	Pluggable Interface = OM622
OP2G5	{1 … 38}	none
OP2G5D	{1 … 48}	none
OP10	{1 … 192}	none
OP10D	{1 … 192}	none
OP40	{1 … 768}	none

The type of HO path termination depends on the interface standard of the related port:

- STS1 for SONET ports
- VC4 for SDH ports

The VC4 path terminations use the same PTNumAIDs as the STS1 path terminations, but only every 4th PTNumAID is valid. The valid PTNumAIDs for the HO path termination types are:

HO Path Termination Type	Valid PTNumAIDs
STS1	1,2,3,4,5,6 …..
VC4	1, 4, 7, 10, …..

Adjunct higher-order path terminations for TransMUX

Higher-order path terminations

Each adjunct port on the asynchronous and synchronous side of the TransMUX provides one path termination.

The type of TransMUX HO path termination is fixed to STS1.

AIDs for higher-order path terminations of asynchronous TransMUX side

The TMuxAPTNumAID shall be used to identify higher-order path terminations of the asynchronous TransMUX Side in the system:

Simple_TMuxAPTNumAID ::= **`1`** | **`all`**

TMuxAPTNumAID ::= TMuxAPortAID–Simple_TMuxAPTNumAID

AIDs for higher-order path terminations of synchronous TransMUX side

The TMuxSPTNumAID shall be used to identify higher-order path terminations of the synchronous TransMUX Side in the system:

Simple_TMuxSPTNumAID ::= **`1`** | **`all`**

TMuxSPTNumAID ::= TMuxSPortAID–Simple_TMuxSPTNumAID

Lower-order tributaries

The LONumAID shall be used to identify lower-order tributaries in the system.

Simple_LONumAID ::= { **1**, … **63** } | **all**

LONumAID ::= PTNumAID-Simple_LONumAID

Legal Simple_LONumAIDs depend on the type of HO path termination as shown in the following table:

HO Path Termination	Legal LO Tributary Numbers	LO Tributary Type
STS1 (SONET)	{1 … 26}	VT1.5
VC4 (SDH)	{1 … 63}	LO VC3, VC12

The LO VC3 tributaries use the same LONumAIDs as the VC12 tributaries, but only every 21st LONumAID is valid. The valid LONumAIDs for the LO tributaries are:

LO Tributary Type	Valid LONumAIDs
VC 12 (SDH)	1,2,3,4,5,6 ….63
LO VC3 (SDH)	1, 22, 43
VT1.5 (SONET)	1,2,3,4,5,6 …. 28

LO VT1.5 tributaries for TransMUX

AIDs for DS1 tributaries

The TMuxSLONumAID shall be used to identify Lower Order Tributaries of the synchronous side of the TransMUX in the system:

Simple_TMuxSLONumAID ::= {1 … 28} | all

TMuxSLONumAID ::= TMuxSPTNumAID–Simple_TMuxSLONumAID

DS1 tributaries

AIDs for DS1 tributaries

The DS1NumAID shall be used to identify the DS1 NIMs in the system:

Simple_DS1NumAID ::= d{1 - 28} | all

DS1NumAID ::= TMuxAPTNumAID–Simple_DS1NumAID

Generic tributary

AIDs for generic tributaries

In some scenarios the specific tributary type is not of interest. The generic tributary is mainly used in the scenarios of cross-connections.

AIDs for generic tributaries

The GenTribAID is used to identify generic tributaries in the system:

GenTribAID ::= STS1NumAID | STS1NumAID | PTNumAID | VCGNumAID

Generic lower-order tributaries

The generic lower-order tributaries are the set of all types of lower-order tributaries. The generic lower-order tributaries can be used in the definition of TL1 commands when the command is applicable for all 3 types of lower-order tributaries.

Generic lower-order Tributary definition:

GenLOTribAID ::= LONumAID | TMuxSLONumAID

Protection groups

Simple protection group AID

The system supports equipment and transmission protection. The ProtGrpAID can be used to identify equipment and transmission protection groups:

Simple_ProtGrpAID ::= **e**EquipmentProtGrpID | **eall**| {**o**|**n**|**t**|**f**} TransProtGrpID | **oall** | **nall**| **tall**| **fall**| **all**

ProtGrpAID ::= BayAID–Simple_ShelfAID–Simple_ProtGrpAID

The first character in the Simple_ProtGrpAID is denoted as *entity type qualifier*. It determines the type of protection as described in the following table:

Entity Type Qualifier	Protection Type
e (quipment)	1+1 or 1:1 (future) equipment protection group
o (ne plus one)	1+1 linear MSP/APS
n (by one)	1:n linear MSP/APS (including 1:1) – future
t (wo fibre)	2 fibre MS SPRING / BLSR
f (our fibre)	4 fibre MS SPRING / BLSR

The *EquipmentProtGrpID* is described in the following subsections.

Transmission protection group ID

The *TransProtGrpID* is a three digit identifier which will be assigned to the group when it is created by the user:

TransProtGrpID ::= { **001**, … , **999** }

Types of transmission protection AIDs

The following types of transmission protection AIDs are defined according to the rules above:

OnePlusOne_ProtGrpAID ::=

BayAID–Simple_ShelfAID–**o**TransProtGrpID | BayAID–Simple_ShelfAID–**oall**

NByOne_ProtGrpAID ::=

BayAID–Simple_ShelfAID–**-n**TransProtGrpID | BayAID–Simple_ShelfAID–**nall**

TwoFibre_ProtGrpAID ::=

BayAID–Simple_ShelfAID–**t**TransProtGrpID | BayAID–Simple_ShelfAID–**tall**

FourFibre_ProtGrpAID ::=

BayAID–Simple_ShelfAID–`f`TransProtGrpID | BayAID–Simple_ShelfAID–**`fall`**

Path protection group AIDs

The output tributary AID of the associated protected cross-connection is used to identify a path protection group AID.

Switch and timing equipment protection group

AID for switch and timing equipment group

The system can have a 1+1 equipment protection group for the XC320 CP which implements the switch and timing function. The SwitchAndTmgProtGrpAID can be used to identify the 1+1 equipment protection group for the XC circuit pack:

Simple_SwitchAndTmgProtGrpAID ::= **estgrp**

SwitchAndTmgProtGrpAID ::= BayAID–Simple_ShelfAID–**estgrp**

Controller equipment protection group

AID for controller equipment group

The system can have a 1+1 equipment protection group for the CTL CP which implements the system controller function (SCF) and data communications function (DCF).The CtlProtGrpAID can be used to identify the 1+1 equipment protection group for the CTL:

Simple_CtlProtGroupAID ::= **ectlgrp**

CtlProtGroupAID ::= BayAID–Simple_ShelfAID–**ectlgrp**

General equipment protection groups

Equipment Protection Group ID

The *EquipProtGrpID* is a two digit identifier which shall be assigned to the group when it is created by the user:

EquipProtGrpID ::= **`{ 01, …, 99 }`**

AID for EP155 equipment group

The *EP155ProtGrpAID* shall be used to identify the 1+1 equipment protection group for the EP155:

Simple_EP155ProtGroupAID::= **`eep`**EquipProtGrpID

EP155ProtGroupAID::=BayAID–Simple_ShelfAID–**`eep`**EquipProtGrpID

AID for EP51 equipment group

The *EP51ProtGrpAID* shall be used to identify the 1+1 equipment protection group for the EP151:

Simple_EP51ProtGroupAID::=**`eep`**EquipProtGrpID

EP51ProtGroupAID::=BayAID–Simple_ShelfAID–**`eep`**EquipProtGrpID

AID for LOXC equipment group

LOXCProtGrpAID shall be used to identify the 1+1 equipment protection group for the LOXC:

Simple_LOXCProtGroupAID::=eloxc1 (1 stands for number one).

LOXCProtGroupAID::=BayAID–Simple_ShelfAID-Simple_LOXCProtGroupAID

AID for TMUX2G5/1 equipment group

The TMUXProtGrpAID shall be used to identify the 1+1 equipment protection group for the TMUX2G5/1:

Simple_TMUXProtGroupAID ::= etmuxEquipProtGrpID

TMUXProtGroupAID ::= BayAID–Simple_ShelfAID–Simple_TMUXProtGroupAID

AID for ES64U equipment group

The ES64UProtGrpAID shall be used to identify the 1+1 equipment protection group for the ES64U:

Simple_ES64UProtGroupAID ::= ees64uEquipProtGrpID

ES64UProtGroupAID ::= BayAID–Simple_ShelfAID–Simple_ES64UProtGroupAID

Examples

AID examples

The following tables gives examples for the various AIDs described in the previous sections:

Entity	Example AID
Bay/Rack	1
Shelf/Subrack	1–1
Equipment Protection Group (Switch and Timing Function)	1–1–estgrp
Equipment Protection Group (Controller)	1–1–ectlgrp
Equipment Protection Group EP155 or EP51	1–1–eep01
Equipment Protection Group LOXC	1–1–eloxc1
Equipment Protection Group TransMUX	1–1–etmux1
Equipment Protection Group ES64U	1–1–ees64u1
1+1 MSP/APS Protection Group	1–1–o100
Two Fiber MS-SPRing/BLSR Protection Group	1–1–t110
Fan Unit	1–1–fan
PI	1–1–pia
TI	1–1–tia
EI	1–1–eci61
Backplane	1–1–backpl
User Panel	1–1–usrpnl
CI–CTL	1–1–cictl
User Byte Port	1–1–ubio3
LAN Ports	1–1–lan2
Slot	1–1–#–#–2
CP	1–1–#–#–xcw–cp
CP (LOXC)	1–1–#–#–39–loxc
Socket	1–1–#–#–5–sc2
NVM	1–1–#–#–ctlw–nvm
SPort	1–1–#–#–1–3
APort	1–1–#–#–21–4
EPort	1–1–#–#–2–1
TMuxAPort	1–1–#–#–15–tma1
TMuxSPort	1–1–#–#–15–tms1

Entity	Example AID
VCG Port	1–1–#–#–2–v1
OTN Port	1–1–#–#–1–ot1
STS–1/AU–3 Trib	1–1–#–#–1–3–1
VCG Trib	1–1–#–#–2–v1–1
HO Path Termination	1–1–#–#–1–16–3
PTF of Asyn. TransMUX Side	1–1–#–#–15–tma1–1
PTF of Syn. TransMUX Side	1–1–#–#–15–tms1–1
LO Tributary	1–1–#–#–1–16–1–63
TransMUX DS1 NIM	1–1–#–#–15–tma1–1–d28
TransMUX VT1.5 PTF	1–1–#–#–15–tms1–1–28

3 TL1 commands - alphabetical order

Overview

Purpose

This section contains the TL1 commands definition in alphabetical order.

Contents

ACT-L1ROUTING

Overview

The command is used to enable the level 1 routing function on an NE and to establish an ONNS enabled node as the proxy of its own domain in the routing hierarchy level 1. The **ACT-L1ROUTING** command can only be executed in maintenance mode.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command syntax:
ACT-L1Routing:*srcTid:srcAid:ctag:RoutingFunc=routeFunc,RoutingAreaID=routeAreaID,ParentRoutingAreaID=prouteAreaID,L1RoutingControllerID=routeControlID,L1RoutingControllerIPAddr=routeControlIPAddr,AbstractModel=abstractModel,AbstractNodeID=abstractNodeID;*

Input parameters

Command parameters

1. **srcTid**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set.
2. **srcAid**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
   Default: snn
3. **ctag**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages. See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string

Range:
The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.

Routing parameters

1. **`routeFunc`**
   routeFunc defines the level 1 routing controller functionality of the addressed ONNS node.
   Type: Alphanumeric value set.
   Range: PRIMARY (ONNS node is the primary level 1 routing controller)
2. **`routeAreaID`**
   routeAreaID defines the identity of the routing area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
3. **`prouteAreaID`**
   prouteAreaID defines the identity of the parent routing area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
4. **`routeControlID`**
   routeControlID defines the identity of the level 1 routing controller.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
5. **`routeControlIPAddr`**
   routeControlIPAddr defines the IP address for this routing controller node as level 1 routing controller address.
   Type: four dot separated numeric string
   Range:
   For first numeric string: 0 … 223
   For second to fourth numeric string: 0 … 255
   The values 0.0.0.0 and 1.1.1.1 are excluded.
6. **`abstractModel`**
   abstractModel defines the abstraction model used by the level 1 routing controller.
   Type: Alphanumeric value set.
   Range: ABSTRACTNODE
7. **`abstractNodeID`**
   abstractNodeID is defined in case the *abstractModel* is ABSTRACTNODE. This parameter specifies the abstract node ID, which is advertised in the level 1 area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string
   Parameter *abstractNodeID* is optional for *abstractModel* values other then ABSTRACTNODE.

Output format

Normal sytem response format to ACT-L1ROUTING

If the ACT-L1ROUTING request accepted by the system, the following normal completion response format below is returned:

```
  srcTid date time
M  ctag COMPLD
;
```

Output parameters

Command parameters

1. **srcTid**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in “RTRV-HDR: Retrieve header” (p. 3-2417).
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **date**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date
3. **time**
   Time. The time parameter is set to the time when the output message is generated.
   See also time in “RTRV-HDR: Retrieve header” (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time

4. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.
5. **`completion code`**
   Standard TL1 completion code result.
   Type: Value set.
   Range:
   - COMPLD (default): for success
   - DENY: in case of errors

Example input/output

Input/Output example to enable a level 1 Routing Controller

The following example shows the successful completion of a ACT-L1ROUTING command to enable an ONNS node as level 1 Routing Controller:

```
ACT-L1ROUTING:LT-ONNS9:snn:113456:RoutingFunc=PRIMARY,RoutingAreaID=111.234.215.6,
ParentRoutingAreaID=111.214.115.7,L1RoutingControllerID=218.3.45.56,
L1RoutingControllerIPAddr=129.32.19.2,AbstractModel=ABSTRACTNODE,
AbstractNodeID=129.239.4.23;

   LT-ONNS9 2002-07-18 08:00:00
M  113456 COMPLD
;
```

Error responses

Validation rule: level 1 Routing controller in place

It is verified that no other node in the level 0 domain has set the attribute *routeFunc* to PRIMARY.

Error code:

`IDNC – Input, Data Not Consistent, level 1 Routing Controller already in place`

Validation rule: Abstract Node ID missing

If *abstractModel* is set to ABSTRACTNODE then the parameter *abstractNodeID* has to be set, too.

Error code: `IPMS – Input, Parameter MiSsing, abstractNodeID is missing`

Validation rule: Check IP configuration

The IPv4 adress for *routeControlIPAddr* has to be checked on the following values:

- for first numeric string: 0 … 223,
- for second to fourth numeric string: 0 … 255,
- the values 0.0.0.0 and 1.1.1.1 are not allowed

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

Validation rule: Check of SCN availability

The routing controller should be only provisionable if the SCN is available on this NE.

Error code: `IDNC – Input, Data Not Consistent, SCN not enabled`

Validation rule: SCN IP address already in use

The SCN IP address of the routing controller has to be different to the SCN IP address (= ASTN node IP address) and the UNI node ID of the NE.

Error code:
`IDNC – Input, Data Not Consistent, IP address of level 1 Routing Controller already in use`

Validation rule: SCN IP address already in use 2

The SCN IP address of the routing controller has to be different to the E-NNI Signaling Controller Protocol Controller SCN address of the NE.

Error code:
`IDNC – Input, Data Not Consistent, RC PC SCN address must be different from SC PC SCN address`

Validation rule: Check of the maintenance condition

The system must be in maintenance condition (MCOND=YES) for change of *routeControlIPAddr*.

Error code: `SNVS – Status, Not in Valid State`

Related TL1 messages

- "CANC-L1ROUTING" (p. 3-54)
- "REPT L1ROUTING notification" (p. 3-2133)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)

ACT-L1ROUTING notification

The ACT-L1ROUTING notification message reports if an ONNS node is enabled as level 1 Routing Controller. Enabling is caused by a user's TL1 command.

Notification messages resulting from a TL1 command that were caused by the user's own requested changes

If an ONNS node is enabled as level 1 Routing Controller as a result of a TL1 ACT-L1ROUTING command, then a REPT DBCHG message is transmitted to all users provisioned to receive REPT DBCHG messages, including the user that issued the command. Therefore, a management OS must expect to receive messages for changes it caused as well as those made by a CIT user.

Output format

The REPT DBCHG message shall use the following format:

```
   sid date time
A  atag REPT DBCHG
   "TIME=chgtm,DATE=chgdat:ccb:[aid]:[com_block]:[spec_block]"
     ...
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **atag** – Automatic Tag
   The *atag* is used for message sequencing. The *atag* is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Type: decimal number.
   Range: 0 to 999.
   Default: 0
2. **chgtm** – Change Time
   Time when the change is occurred.
   Type: value set.
   Range: HH-MM-SS.
3. **chgdat** – Change Date
   Date when the change occurred
   Type: Value set.
   Range: YY-MM-DD

4. **`ccb`** – Command Code Block
   This field will be set to the TL1 command that causes that database change. The *com_block* and *spec_block* fields will be set according to the parameters that changed.
   Type: Alphanumeric value set verb-modifier[-modifier].
5. **`aid`** – Access Identifier
   AID of the entity for which the change is reported
   Type: AID.
   Range: As defined by the *aid* input parameter of the corresponding command, defined by *ccb*.
6. **`com_block`** – Common Block
   This is a position-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter values are included in this block in the same order as the corresponding TL1 command.
   As exception to this rule are some commands in the protection fragment, where the aliases *rid* and *ppgname* are not reported in the **`REPT DBCHG`** notification.
   If the database change is the result of a TL1 command, then the parameter *ctag* from the TL1 command is not included in the *com_block*.
   Type: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
7. **`spec_block`** – Specific Block
   This is a name-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter names and values are included in this block. Parameters within the specific block are positionally independent and are specified using a name-defined construct of *parameter* = value in a comma-separated list. The parameter names are the same as those for the corresponding TL1 command.
   Type: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.

Spec block parameters

1. **`RoutingFunc`**
   routeFunc defines if the *LSAAdvertisedRouter* has level 1 routing controller functionality.
   Type: Alphanumeric value set.
   Range: PRIMARY (ONNS node is the primary level 1 routing controller)
2. **`RoutingAreaID`**
   routeAreaID defines the identity of the routing area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
3. **`ParentRoutingAreaID`**
   prouteAreaID defines the identity of the parent routing area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
4. **`RoutingControlID`**
   routeControlID defines the identity of the level 1 routing controller.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string.
5. **`RoutingControlIPAddr`**
   routeControlIPAddr defines the IP address for this routing controller node as level 1 routing controller address.
   Type: four dot separated numeric string
   Range:
   For first numeric string: 0 … 223
   For second to fourth numeric string: 0 … 255
   The values 0.0.0.0 and 1.1.1.1 are excluded.
6. **`AbstractModel`**
   abstractModel defines the abstraction model used by the level 1 routing controller.
   Type: Alphanumeric value set.
   Range: ABSTRACTNODE
7. **`AbstractNodeID`**
   abstractNodeID is defined in case the *abstractModel* is ABSTRACTNODE. This parameter specifies the abstract node ID, which is advertised in the level 1 area.
   Type: four dot separated numeric string
   Range: 0 – 255 for each numeric string
   Parameter *abstractNodeID* is optional for *abstractModel* values other then ABSTRACTNODE.

Detailed behavior description

The **`REPT DBCHG`** message reports any change in the system database. Changes can be caused by a TL1 user or by an internally-generated state change.

Changes in user provisionable attributes are reported as the equivalent TL1 command that would cause that change.

Changes in non-user provisionable attributes are reported using only the AID and the block parameters, i.e. leaving the *ccb* parameter empty.

System initialization

Any time a command is entered that causes a system reset, a **REPT DBCHG** message is transmitted before the system resets.

AID ranges

If a database change occurs as a result of a TL1 command in which an AID range is entered, the system responds separate **REPT DBCHG** messages for each affected AID.

Messages caused by own changes

If the database change occurs as a result of a TL1 command, a **REPT DBCHG** message is transmitted to users provisioned to receive **REPT DBCHG** messages, including the user which issued the command.

Example input/output

The following is an example of the **REPT DBCHG** command:

```
   UNITE-NE 00-11-08 08:00:00
A  001 REPT DBCHG
   ″TIME=07-59-59,DATE=00-11-08: ENT-EQPT:1-1-#-#-05::nvmname=OP2G5,nvmqual=13SR4″
;
```

The following example shows the output of a REPT L1ROUTING notification caused by a ACT-L1ROUTING TL1 command:

```
   LT-ONNS9 2002-07-18 08:00:00
A  113456 REPT DBCHG
   ″TIME=07-59-59,DATE=2002-08-01:ACT-L1ROUTING:::RoutingFunc=PRIMARY,
RoutingAreaID=111.234.215.6,ParentRoutingAreaID=111.214.115.7,
L1RoutingControllerID=218.3.45.56,L1RoutingControllerIPAddr=129.32.19.2,
AbstractModel=ABSTRACTNODE,AbstractNodeID=129.239.4.23″;
```

References

GR-199-CORE

ACT-USER: Activate user

The **ACT-USER** command must be initiated to set up a login session with the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): Any

Priority: 4

Abortable: No

Input format

Command syntax: **ACT-USER**:*tid*:*uid*:*ctag*::*pid*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **uid** – User Identifier
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 5 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
   Default: not applicable
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **pid** – Password Identifier
   This required parameter is the confidential password authenticator for the given *uid*. Password strings are transmitted in non-encrypted form in the **ACT-USER** command; are encrypted when stored in the NE; and are never transmitted from the NE.
   Type: character string (case-sensitive)
   Range: 6 to 10 legal characters. Valid passwords consist of at least two non-alphabetic characters with at least one special character. The special character can be one of the required non-alpha characters. For example, TEST1+ is a valid password. The first character of a password must be a letter.
   Default: For the two superusers the default password is: UNITE+01 for User ID LUC01 and UNITE+02 for User ID LUC02. For other users: not applicable.

Output format

Normal Completion Message without Customer Proprietary Banner

When a user logs in successfully while *no* customer proprietary has been provisioned, then the following response is returned:

```
   sid date time
M  ctag COMPLD
   "uid:lastdate,lasttime,attempts,systype,release,upc:spec_block"
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release release
   User Privilege Code: upc
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENT AND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */

;
```

Normal Completion Message with Customer Proprietary Banner

When a user logs in successfully while *a* customer proprietary has been provisioned, then the following response is returned:

```
   sid date time
M  ctag COMPLD
   "uid:lastdate,lasttime,attempts,systype,release,upc:spec_block"
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release release
   User Privilege Code: upc
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENT AND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */
   /*
   <LINE1>
   <LINE2>
    ...
    ...
    ...
   <LINE10>
```

```
   <LINE11>
   */

;
```

Expired password

If the login request would otherwise complete successfully, but the user password has expired or it is the first login, the following partial completion response is returned:

```
   sid date time
M  ctag PRTL
   "uid:lastdate,lasttime,attempts"
   /* Your password has expired. Until you change your password (ED-PID)
      you will not be allowed further access to this Network Element.
   */
;
```

Advance notice of password expiration

If the user password is about to expire, the following line is added at the bottom of the login banner, where 'x' is a number from 1 to 5:

```
   ...
   Your password will expire in x days.
   */
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **RTRV-HDR** command.

Output parameters

Command parameters

1. **uid** – User Identifier
   This is included in the command by the OS/CIT and repeated by the NE as a confirmation that the given user identifier is successfully logged in.
2. **lastdate** – Last uid Login Date
   This is the date of the last session established by this *uid*.
   Type: value set of the format YY-MM-DD
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
   Default:
   Date of the User Id creation for the first login after the User Id has been created
   Date of the User Id enabling for the first login after the User Id was disabled.

3. **`lasttime`** – Last uid Login Time
   This is the time of the last session established by this *uid*.
   Type: value set of the format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents the hours, MM the minutes, SS the seconds. The 24 hour format is used.
   Default:
   Date of the User Id creation for the first login after the User Id has been created
   Date of the User Id enabling for the first login after the User Id was disabled.
4. **`attempts`** – Unsuccessful Login Attempts
   This is the number of unsuccessful login attempts since the last successful login session on this ne.
   Type: Integer.
   Range: 0 – 98
5. **`systype`** – System Type
   This specifies the type of system.
   Type: Value set.
   Range: WaveStar_UNITE
   Default: WaveStar_UNITE
6. **`release`** – Release
   This is the software Release number of the system.
   Type: value set in the form xx.yy.zz
   Range: x, y, z are digits. Leading zeros are not suppressed. The version number ranges from 01.00.00 … 99.99.99
   Example: 01.02.03
   Default: 01.00.00
7. **`upc`** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation "&" is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users:
   P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also "List of user privilege codes" (p. 15-1).

8. **line1** – Line 1 of the Customer Provisionable Banner.
   Setting *line1* to an empty string will suppress the output of the Customer Provisionable Banner in the response to the **ACT-USER** command.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.
9. **line2 … line11** – Line 2 … Line 11 of the Customer Provisionable Banner.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.

Spec block parameters

1. **sysavst** – System Availability Status
   Because logins are only possible after the system has finished the recovery, the only possible value of sysavst is AVAILABLE.
   Type: Alphanumeric value set.
   Range: AVAILABLE
2. **tmleft** – Time left before password expires
   Time left before password expires. If there are more than 5 days left before the password expires, this parameter is omitted.
   Type: Integer.
   Range: 1 … 5

Detailed behavior description

An active TL1 access login session is required at each NE in order to interact with that NE (including receiving autonomous message output as well as using TL1 commands).

A maximum of one login per active session is allowed. Any subsequent attempt on the same session is denied. But the user can login several times on different sessions.

Until a successful login attempt is complete for a given NE, there is no communication (autonomous or command responses) outbound from the NE except to DENY unsuccessful login attempts.

If a user password has expired upon execution of the **ACT-USER** command, the ACT-USER will be accepted but that user will not be able to perform any function or receive any autonomous message output until the associated password has successfully been modified. This will also occur if the user logs in the first time.

A special banner message will be displayed as part of the command completion response, informing the user that the password has expired and must be updated (see the OUTPUT FORMAT section). The user is allowed to use either ED-PID to update the password identifier or CANC-USER to terminate the session.

When the NE receives an ACT-USER command, the given login ID will initiate a login session on the NE provided that a provisioned login ID and correct password are entered.

When the communication with the NE fails, all active logins to that NE are terminated without notice to the user. Similarly, all active logins to a targeted NE are terminated if a communication failure occurs between the local NE and that remote NE or if an intermediate NE is reset or initialized.

The **ACT-USER** command does not generate a **REPT DBCHG** message. Each ACT-USER TL1 command is entered in the Security Information Log.

The Customer Provisionable Banner is displayed only if LINE1 of the Customer Provisionable Banner is not an empty string; otherwise the Customer Provisionable Banner is omitted.

Example input/output

The following example shows a successful login session initiation attempt with no Customer Provisionable Banner.

ACT-USER:UNITE-123:nbguser1:123456::kj5lee+3;

```
   UNITE-123 07-12-21 08:00:00
M  123456 COMPLD
   ”nbguser1:07-12-20,09-00-00,1,WaveStar_UNITE,10.00.00,P3&M3:sysavst=AVAILABLE”
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release 10.00.00
   User Privilege Code: P3&M3
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENTAND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */

;
```

The following example shows a partial completion response to a login session initiation attempt:

ACT-USER:LT-UNITE-123:nbguser1:123456::kj5lee+3;

```
   LT-UNITE-123 01-01-01 08:00:00
M  123456 PRTL
   ”nbguser1:00-12-31,09-00-00,23”
   /* Your password has expired. Until you change your password (ED-PID)
    you will not be allowed further access to this Network Element */
```

;

Note: A blank line in front of the terminator ″;″ can be ignored.

Error responses

Validation rule: Maximum logins exceeded

If a user attempts to login after the maximum number of login sessions has been reached, the following error message will be displayed.

Error code: `SARB – Status, All Resources Busy`

Validation rule: Invalid uid and/or pid

If the NE receives an ACT-USER command with an invalid *uid* (*uid* not in the NE database or *uid* syntactically incorrect), invalid *pid* or both.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Disabled User ID

If a user attempts to log in using a disabled User ID, the login is denied and the following error message will be displayed:

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Expired password and unsuccessful change of password

The user will be allowed three attempts to change their password. If the third attempt fails, the user will be logged out and the association dropped. The execution of any command prior to the successful change of the password will be counted as one of the three attempts to change the password. This includes failures of the **ED-PID** command due to incorrect data, attempts to execute valid commands other than the **ED-PID** command, or invalid commands, possibly due to a data entry error. In all cases, the system will return the following error response.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Login on established session

If the system receives an ACT-USER command on an already established session the system responds with the following error response:

Error code: `ICNV – Input, Command Not Valid`

Validation rule: Maximum number of user sessions via T-TD on Gateway NE

If a user session is established via T-TD on a Gateway NE with CTL/- or CTL/2 with 20 TCP sessions and while already 300 user sessions are existing through the T-TD, then the following error response is sent:

Error code:

`SSRE – Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`

This error response is sent by the Gateway NE, but contains the TID from the ACT-User command.

Validation rule: Maximum number of user sessions via T-TD on Gateway NE with CTL/4

If a user session is established via T-TD on a Gateway NE with CTL/4T or CTL/4S with 20 TCP sessions and while already 300 user sessions are existing through the T-TD, then the following error response is sent::

Error code:

`SSRE – Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`

This error response is sent by the Gateway NE, but contains the TID from the ACT-User command.

Validation rule: Maximum number of user sessions via T-TD on Gateway NE

If a user session is established via T-TD on a Gateway NE with CTL/3 with 20 TCP sessions and while already 600 to 603 user sessions are existing through the T-TD, then the following error response is sent:

Error code:

`SSRE – Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`

This error response is sent by the Gateway NE, but contains the TID from the ACT-User command.

Validation rule: maximum number of user sessions via T-TD on Gateway NE per TCP connection

If a user session is established via T-TD on a Gateway NE using a TCP connection on which already 300 user sessions are existing, then the following error response is sent:

Error code:

`SSRE – Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`

This error response is sent by the Gateway NE, but contains the TID from the ACT-User command.

Related TL1 messages

- "CANC-USER: Cancel user" (p. 3-60)
- "CANC-USER-SECU: Cancel user security" (p. 3-62)
- "ED-PID: Edit password identifier" (p. 3-798)
- "DLT-USER-SECU: Delete user security" (p. 3-272)
- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "ENT-USER-SECU: Enter user security" (p. 3-1970)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)
- "RTRV-PRMTR-SFTWR: Retrieve parameter software" (p. 3-2753)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

ALW-MSG: Allow message

The **ALW-MSG** command allows the NE to resume the forwarding of autonomous messages previously suspended with the **INH-MSG** command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: no

Input format

Command syntax: **ALW-MSG-ALL**:*tid*::*ctag*[::,,,,*sessions*];

The purpose of the commas in case the optional parameter *sessions* is present is that the format is compliant with non-supported position defined parameters in GR-833.

Input parameters

Command parameters

1. **tid** – Target Identifier
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **sessions** – Affected sessions
Refer to the definition of output parameter *sessions* in "RTRV-ATTR-MSG: Retrieve attribute message" (p. 3-2260).

Output format

If the ALW-MSG request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command enables autonomous message reporting for *sessions*.

If this command is executed when autonomous messages are already being transmitted, it is complete successfully.

Example input/output

The following is an example of the **INH-MSG** command:

```
ALW-MSG-ALL:LT-UNITE::123456;

   LT-UNITE 00-11-15 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ALW-MSG** command.

Related TL1 messages

- “INH-MSG: Inhibit message” (p. 3-2021)
- “RTRV-ATTR-MSG: Retrieve attribute message” (p. 3-2260)

References

GR-833-CORE

APPLY: Apply software generic or activate a downloaded database

The **APPLY** command can be used to

- install a copy of a software generic stored in the NE, overwriting the currently executing software. Installation of the software causes the system to reset.
- activate a previously downloaded database. Activation causes the system to reset.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 2

Abortable: No

Input format

Command syntax: **APPLY**:*tid*::*ctag*::*[,,qualifier]*;

The two commas before *qualifier* are kept for compatibility reasons for products, for which scheduling of an APPLY command is possible.

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **qualifier**

   Description: The *qualifier* selects the software generic to be installed.

   - INSTALL causes the software download to be installed. This is the default action if no value is specified for *qualifier*. Prior to installation, the software state must be equal to DOWNLOADED. The software state can be retrieved by **RTRV-PRMTR-SFTWR**.
   - TRIAL causes the downloaded software to be installed in a trial phase. During the trial phase, it is possible to switch back to the previously installed software and database. Prior to installation, the software state must be equal to DOWNLOADED. The software state can be retrieved by **RTRV-PRMTR-SFTWR**.

- COMMIT terminates the trial phase of a newly installed software. When the trial phase is terminated, a switch back to the previously installed software and database is no longer possible. The termination of the trial phase is allowed with any state of the previous SW generic and the previous database.
- REVERT forces the system to switch back to the previously installed software and database. The switch back is only possible during trial phase. All provisioning done during the trial phase will be lost. Prior to the switch back, the state of the previous SW generic and the state of the previous database must be PRESERVED. The software and database states can be retrieved by **RTRV-PRMTR-SFTWR** and **RTRV-PRMTR-DATA**.
- DATABASE causes the downloaded database to be installed as working database. Prior to installation, the database state in the previous NVM must be equal to DOWNLOADED.

  The database state can be retrieved by **RTRV-PRMTR-DATA**. The **APPLY** command with the qualifier DATABASE shall compare the system backplane serial number with the serial number stored in previously downloaded database. The command is only executed, if the serial numbers are equal.
- DBFORCED causes the downloaded database to be installed as working database. Prior to installation, the database state in the previous NVM must be equal to DOWNLOADED or MISMATCH.

  The database state can be retrieved by **RTRV-PRMTR-DATA**.

  The **APPLY** command with the qualifier DBFORCED **does not** overwrite the serial number in the downloaded database with the system serial number.

  After the **APPLY** of the new database, the system will start up in maintenance mode where the operator needs to commit to overwrite the serial number of the database with the system serial number.

  The purpose of the qualifier DBFORCED is to install a database which has been backed up from another system, for example, to replace a complete system (including the subrack).

Type: Alphanumeric value set.
Range: INSTALL, TRIAL, COMMIT, REVERT, DATABASE, DBFORCED.
Default: INSTALL.

Output format

Normal completion message

If the NE fully complies with the **APPLY** command successfully, then the following completion response is given:

```
   sid date time
M  ctag COMPLD
   /* New software generic installation in progress */
;
```

Completion message if ES64U databases missing or superfluous

If ES64U databases are missing or superfluous in the database generic and the user has applied the generic with DBFORCE, then the following completion response is given:

```
   sid date time
M  ctag COMPLD
   /* ES64U databases for following slots are missing or superfluous:<List>*/
;
```

The <List> shall be a simple list of slot-numbers separated by commas.

Completion message if previous ES64U databases cannot be restored

If the command with qualifier REVERT could not restore the previous databases for all ES64U packs in the system, then the following completion response is given:

```
   sid date time
M  ctag COMPLD
   /* Previous ES64U databases for following slots are not restored:<List>*/
;
```

The <List> shall be a simple list of slot-numbers separated by commas.

Detailed behavior description

APPLY being denied during installation

If an installation is already in progress, then another request for an installation will be denied.

APPLY during duplex control recovery

A request for installation will be denied if a second system controller is in recovery.

Indication of COMPLD

The **APPLY** command shall report COMPLD before the command causes the system to be reset. When the system is reset, the DCN communication will fail. The command will indicate completion before that occurs. This does not guarantee, that the COMPLD indication is sent successfully. The COMPLD indication may be lost.

Generation of REPT DBCHG message

The **APPLY** command generates a **REPT DBCHG** message. This message will be generated before the command causes the system to be reset. This does not guarantee, that the REPT DBCHG message is sent successfully. The REPT DBCHG message may be lost.

Installation recorded in history log

The initiation of an installation is recorded in history log.

Reactions after installation completed

When a NE receives an **APPLY** command and as a result successfully installs a new software generic, the NE loses any command messages that were pending and all alarms are cleared. All performance monitoring bins are cleared, and any conditions are cleared.

When a NE receives an **APPLY** command and as a result successfully installs a downloaded database, the NE loses any command messages that were pending and performance monitoring bins are cleared are cleared.

The system performs a CTL controller reset to finalize the installation.

SW Generic Installation with Trial Phase

The qualifier value TRIAL shall prepare the installation of the downloaded SW generic in trial phase. With the trial phase, the user keeps the option to switch back to the previous software and database. The trial phase must be explicitly invoked with qualifier TRIAL. If the qualifier is omitted, or INSTALL, the system does not enter the trial phase and a switch back to the previous software is not possible.

SW Generic Running in Trial Phase - Software and database state during trial phase

During trial phase, the software and database state is:

- SW state in current NVM: RUNNING
- SW state in previous NVM: PRESERVED
- DB state in current NVM: RUNNING
- DB state in previous NVM: PRESERVED.

Any other combination of state values is not considered as trial phase. If the system detects during start up that only one of the previous NVM states is PRESERVED, then the state PRESERVED is modified to PREVIOUS.

SW Generic Running in Trial Phase - Commands not allowed in trial phase

As long as the software is running in trial phase, the system shall not allow to execute commands that will make a switch back to the previous software impossible. The commands not allowed in trial phase are:

- Software Download via CPY-MEM or COPY-RFILE
- Database Restoration via CPY-MEM or COPY-RFILE
- Activation of ONNS with ED-EQPT (System).

SW Generic Running in Trial Phase - Termination of the trial phase

The qualifier value COMMIT shall terminate the trial phase. The SW state and DB state in previous NVM is set to PREVIOUS. After the trial phase is terminated, the system shall allow the exection of TL1 commands for Software Download, Database Restoration and ONNS activation.

An attempt to terminate a trial phase shall always succeed, independent, whether the system is in trial phase or not. If the system is not in trial phase, and either the SW state or the DB state of the previous NVM is still PRESERVED, then the system shall only set this state from PRESERVED to PREVIOUS.

The SW is in trial phase, as long as the SW and the DB state of the previous NVM is PRESERVED. If only one of the states is PRESERVED, while the other is PREVIOUS, the trial phase is assumed to be terminated. When the **APPLY** command with the qualifier COMMIT is executed, the remaining PRESERVED state is also changed to PREVIOUS. This clean-up action removes the inconsistent SW or DB state.

SW Generic Running in Trial Phase - Switch back to the previous software and database

As long as the newly installed software is in trial phase, the system shall allow to switch back to the previously installed software and database. The qualifer value REVERT forces the system to swap current and previous NVM for software and database.

After the switch back to the previous software and database has been performed, the software and database state is:

- SW state in current NVM: RUNNING
- SW state in previous NVM: PREVIOUS
- DB state in current NVM: RUNNING
- DB state in previous NVM: PREVIOUS.

With the switch back to the previous software and database, any provisioning done during the trial phase will be lost.

Switch back to the previous software and database for ES64U

SW fallback is fully traffic affecting for the ES64U.

Example input/output

The following example shows an APPLY command requesting installation of a new generic:

```
APPLY:LT-UNITE::123456;
   LT-UNITE 00-11-01 09:45:28
M  123456 COMPLD
   /* New Software generic installation in progress */
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command error response section. The error responses listed there also apply to the **APPLY** command. Additional error responses that specifically apply to this command are defined in the following:

Validation rule: Temporary system exhaustion

If the system is experiencing temporary exhaustion of allocated resources, the system responds with:

Error code: `SARB – Status, All Resources Busy, system limit exceeded`

Validation rule: APPLY already in progress

If an APPLY request is issued and an installation is already in progress or a resource is busy for other reasons, the system responds with:

Error code: `SARB – Status, All Resources Busy`

Validation rule: Non-Hardware system difficulties

If the command cannot be completed due to system difficulties other than hardware or equipment, the system responds with:

Error code: `SROF – Status, Requested Operation Failed`

The following list identifies conditions that will cause an SROF error response:

- Install failure

Validation rule: Generic not available

If the command failed because the generic is not available, the following error response is returned:

Error code: `ENSG – Equipage, Not Software Generic, generic not available`

Validation rule: APPLY applied on corrupted software

If an APPLY request is entered with qualifier INSTALL or TRIAL the state of the download software is different from DOWNLOADED, then the following error response is given:

Error code:
`SNVS – Status, Not in Valid State, Software generic corrupt or not available`

This applies for the following software states: CORRUPT, INITIAL, INPROGRESS, PREVIOUS, PRESERVED and RUNNING.

Validation rule: Second APPLY on same software

If an APPLY request is entered and the software copy has been previously installed (that is, its state is PREVIOUS), then the following error response is given:

Error code: `SNVS – Status, Not in Valid State`

Validation rule: No SW installation during DB backup, DB download or DB restore

The system denies an attempt to install a new or previos SW generic, i.e. qualifier is INSTALL, TRIAL, REVERT or omitted, while a DB backup, DB download, or DB restore is in progress.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: No SW installation during Duplex Control Recovery

The system rejects an attempt to APPLY a new software while the Duplex Control Protection Group is in recovery with the following error code:

Error code:
`SNVS – Status, Not in Valid State. Duplex Control Recovery in progress`

Validation rule: No DB activation during Duplex Control Recovery

The system shall reject an attempt to APPLY a downloaded database while the Duplex Control Protection Group is in recovery with the following error code:

Error code:
`SNVS – Status, Not in Valid State, Duplex Control Recovery in progress`

Validation rule: File inappropriate

If the type of file is inappropriate for installation to the NE, then the following error response is given:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: SW installation in trial phase while ONNS is active

Execution of the command with qualifier = TRIAL is not allowed if the parameter xcsetup has the value ″SWISS″.

Error code: `SNVS – Status, Not in Valid State, ONNS is active`

Validation rule: Switch back to previous SW only during trial phase

The switch back to the previously installed software and database is only allowed while the system is in trial phase, i.e. the software has been installed using the qualifer value TRIAL and the trial phase has not been terminated with COMMIT.

Error code:
`SNVS – Status, Not in Valid State, switch back to previous software only allowed in trial phase`

Validation rule: Second APPLY on previous database

If an APPLY request is entered with qualifier DATABASE or DBFORCED and the database has been previously installed (that is, its state in the previous NVM is PREVIOUS), then the following error response is given:

Error code: `SNVS – Status, Not in Valid State, Database already installed`

Validation rule: APPLY applied on corrupted database

If an APPLY request is entered with qualifier DATABASE or DBFORCED and the database in the related NVM is in a state different from DOWNLOADED and PREVIOUS, then the following error response is given:

Error code:

`SNVS – Status, Not in Valid State, database corrupt or not available`

Validation rule: Serial number in the database not equal to system backplane serial number

If an APPLY request is entered with the qualifier DATABASE and the serial number stored in the downloaded database does not match the system backplane serial number, then the following error response is given:

Error code:

`SNVS – Status, Not in Valid State, Backplane serial number mismatch`

Validation rule: NVM contains DB for NOT existing ES64U pack

If an **`APPLY`** request is entered with qualifier DATABASE and the database generic contains a database for a not existing ES64U AND the validation rule “NVM NOT contains DB for a existing ES64U Pack” does not apply , then the following error response is given:

Error code:

`SNVS – Status, Not in Valid State, NVM contains DB for not existing ES64U packs`

Validation rule: NVM NOT contains DB for existing ES64U packs

If an **`APPLY`** request is entered with qualifier DATABASE **and** the database generic does **not** contain databases for existing ES64U packs, then the following error response is given:

Error code:

`SNVS – Status, Not in Valid State, NVM contains no DB for existing ES64U packs`

Validation rule: Not all databases of ES64U preserved for trial phase

If the command is executed with *qualifier* = TRIAL in a system with ES64U packs provisioned, then the command shall only successfully complete if the databases of all provisioned ES64U packs can be preserved. The command is denied if not all ES64U databases can be preserved.

Error code:

`SROF – Status, Requested Operation Failed, not all ES64U databases preserved`

Related TL1 messages

- "CPY-MEM: Copy memory" (p. 3-114)
- "RTRV-PRMTR-DATA: Retrieve parameter data" (p. 3-2744)
- "RTRV-PRMTR-SFTWR: Retrieve parameter software" (p. 3-2753)

APPLY notification

Syntax

The APPLY notification has the following specific syntax:

Command syntax: **`APPLY`**:,,*qualifier*;

The output parameters *sid*, *date*, *time*, and *atag* are described in the REPT DBCHG notification.

Output parameters

Command parameters

1. **`qualifier`**

   Description: The *qualifier* selects the software generic to be installed.

   - INSTALL causes the software download to be installed. This is the default action if no value is specified for *qualifier*. Prior to installation, the software state must be equal to DOWNLOADED. The software state can be retrieved by **`RTRV-PRMTR-SFTWR`**.
   - TRIAL causes the downloaded software to be installed in a trial phase. During the trial phase, it is possible to switch back to the previously installed software and database. Prior to installation, the software state must be equal to DOWNLOADED. The software state can be retrieved by **`RTRV-PRMTR-SFTWR`**.
   - COMMIT terminates the trial phase of a newly installed software. When the trial phase is terminated, a switch back to the previously installed software and database is no longer possible. The termination of the trial phase is allowed with any state of the previous SW generic and the previous database.
   - REVERT forces the system to switch back to the previously installed software and database. The switch back is only possible during trial phase. All provisioning done during the trial phase will be lost. Prior to the switch back, the state of the previous SW generic and the state of the previous database must be PRESERVED. The software and database states can be retrieved by **`RTRV-PRMTR-SFTWR`** and **`RTRV-PRMTR-DATA`**.
   - DATABASE causes the downloaded database to be installed as working database. Prior to installation, the database state in the previous NVM must be equal to DOWNLOADED.

     The database state can be retrieved by **`RTRV-PRMTR-DATA`**. The **`APPLY`** command with the qualifier DATABASE shall compare the system backplane serial number with the serial number stored in previously downloaded database. The command is only executed, if the serial numbers are equal.
   - DBFORCED causes the downloaded database to be installed as working database. Prior to installation, the database state in the previous NVM must be equal to DOWNLOADED.

     The database state can be retrieved by **`RTRV-PRMTR-DATA`**.

The **APPLY** command with the qualifier DBFORCED **does not** overwrite the serial number in the downloaded database with the system serial number.

After the **APPLY** of the new database, the system will start up in maintenance mode where the operator needs to commit to overwrite the serial number of the database with the system serial number.

The purpose of the qualifier DBFORCED is to install a database which has been backed up from another system, for example, to replace a complete system (including the subrack).

Type: Alphanumeric value set.
Range: INSTALL, TRIAL, COMMIT, REVERT, DATABASE, DBFORCED.
Default: INSTALL.

Example output

The following is an example for an **APPLY** notification using the qualifier INSTALL:

```
   LT-UNITE 00-11-01 15:00:00
A  000 REPT DBCHG
   ”TIME=15-00-00,DATE=00-11-03:APPLY::,,INSTALL”
;
```

AUD: Perform audit data

Overview

The **AUD**-mod command requests ONNS data to be compared either with local host NE or with the rest of network for audit. Any discrepancies in data is notified as a result in the response. The audit results provide users with information to perform synchronization (either manually or automatically by NN) to improve overall integrity of NN data.

The targeted NN data for audit process are cross-connect map, port object, and path data. The modifier of the **AUD**-mod command indicates the scope of data to be checked.

AUD-LOCAL

AUD-LOCAL triggers local data-audit that is to identify any differences in local NN data from host NE data with respect to port object and cross-connect map. Depending on the result, if any discrepancies found, user can **ED-SYNCDATA-LOCAL** command to request automated data synchronization process by ONNS.

The **AUD-LOCAL** command requests to perform an audit for local host/NN data comparison.

AUD-PATHNBR

AUD-PATHNBR triggers path neighbor data-audit that is to identify any differences in path data kept locally with the neighboring nodes. NN will exchange the current path information with its neighboring NN nodes (in terms of transport network topology) about the NN paths that traverse through this node. For this cross multiple-NE process, NN provides only data-audit without automated process for synchronization. However, based on the result of data-audit, user can fix pathID via **ED-SYNCDATA-PATH** command only if the correct value is clearly known.

The **AUD-PATHNBR** command requests to perform an audit for local path data with neighbor node path data comparison.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No

Input format

Command syntax:

AUD-LOCAL:*srcTid:srcAid:ctag;*

AUD-PATHNBR:*srcTid:srcAid:ctag;*

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

Output format

The output shows any mismatched port objects and mismatched cross connect information in comparing NN data and host NE data.

For mismatched port objects, the cases may occur are as follows:

- port object exists only in NN data, but not in host NE data (ExtraPortIDs)
- port object exists only in host NE data, but not in NN.

The cases in the response represent the mismatched port objects or mismatched cross connect information as follows:

- ExtraPortIDs – port object exists only in NN data, but not in host NE data
- MissingPortIDs – pport object exists only in host NE data, but not in NN data
- ExtraNNXcons – NN owned cross-connect exists only in NN data, but not in host NE
- MissingNNXcons – NN owned cross-connect exists only in host NE data, but not in NN data

- ExtraNonNNXcons – non-NN owned cross-connect exists only in NN data, but not in host NE
- MissingNonNNXcons – non-NN owned cross-connect exists only in host NE data, but not in NN data.

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

The output shall show any mismatched path data in comparing path information at a local NN with its neighboring nodes.

The cases in the response represent the mismatched path information as follows:

- When all the parameters have valid values – local node and neighboring node have different pathIDs for the same path.
- When the parameter for pathID at neighboring node (nbrpathID) is empty – local node has a cross-connect that is connected to a neighboring node, but the neighboring node does not have the corresponding cross-connect or path information.
- When the parameter for pathID at local node (localpathID) is empty – neighboring node has a cross-connect that is connected to a local node, but the local node does not have the corresponding cross-connect or path information.

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

All neighbor nodes which are not reachable are be listed. For these neighbor nodes is was not possible to retrieve information for comparison of the ONNS path data.

Output parameters

Command parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the section "Output parameters" of the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) command.

1. **pathID**
   The PathID of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0…255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65111
2. **localpathID**
   The PathID of the local node of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0…255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65535
3. **nbrpathID**
   The PathID of the neighbor node of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0…255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65111
4. **nbrnodeID**
   ONNS node Network Access identifier (nodeID) of the neighbor node. *nodeID* is not changable and may be different from Signaling network IP address.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0…255.
5. **bandwidth**
   ONNS bandwidth is equal to or greater than the minimum cross connect rate (minXCrate) of the port type. Note: Only SONET switching levels in enumeration. For actual cross-connections ONNS adapts to SDH formats automatically if the ingress port is SDH.
   Values: STS1, STS3, STS12, STS48, STS192.
6. **portID**
   portID of port object discrepancy between NN data and NE data. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # . ONNS assumes a bidirectional port pair.
   Type: dash-separated list of 5 numeric strings.

Example: 1-1-3-1-4
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##. This range is valid for protecting connections in M:N connection groups

7. **connPtID1**
ONNS connection point identifier (connPtID1) for the ingress RX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame #(1), Shelf #(1), Slot/Pack ##, Port ##, Tributary ##
Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
Range: Slot/Pack #*, Port #*, Tributary #*
Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
Range:Slot/Pack #@, Port #@, Tributary #@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

8. **connPtID2**
ONNS connection point identifier (connPtID2) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame #(1), Shelf #(1), Slot/Pack ##, Port ##, Tributary ##
Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
Range: Slot/Pack #*, Port #*, Tributary #*
Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
Range:Slot/Pack #@, Port #@, Tributary #@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

9. **connPtID3**
   ONNS connection point identifier (connPtID3) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
   Type: dash-separated list of 5 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
   Range: Bay/Frame #(1), Shelf #(1), Slot/Pack ##, Port ##, Tributary ##
   Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
   Range: Slot/Pack #*, Port #*, Tributary #*
   Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
   Range:Slot/Pack #@, Port #@, Tributary #@
   Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.
10. **connPtID4**
   ONNS connection point identifier (connPtID4) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
   Type: dash-separated list of 5 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
   Range: Bay/Frame #(1), Shelf #(1), Slot/Pack ##, Port ##, Tributary ##
   Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
   Range: Slot/Pack #*, Port #*, Tributary #*
   Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
   Range:Slot/Pack #@, Port #@, Tributary #@
   Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

Example input/output

The following example shows a successful completion of a **AUD-LOCAL** command for local ONNS data audit operation by the NE with all data in-sync:

```
AUD-LOCAL:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″ExtraPortIDs=NONE″
   ″MissingPortIDs=NONE″
   ″ExtraNNXcons=NONE″
   ″MissingNNXcons=NONE″
   ″ExtraNonNNXcons=NONE″
   ″MissingNonNNXcons=NONE″
   ″MultipleNNXcons=NONE″
;
```

The following example shows a successful completion of a **AUD-LOCAL** command for local ONNS data audit operation by the NE with several data not in sync:

```
AUD-LOCAL:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″ExtraPortIDs=1-1-20-3-5″
   ″MissingPortIDs=NONE″
   ″ExtraNNXcons=1-6-5-3-1,-X-,1-2-4-3-1,sts48,29.13.196.203-43216
   :1-1-4-4-1,-X-,1-2-1-3-1,sts12,29.13.18.23-12345″
   ″MissingNNXcons=1-1-8-10-2,-X-,1-1-3-3-4,sts48″
   ″ExtraNonNNXcons=NONE″
   ″MissingNonNNXcons=NONE″
   ″MultipleNNXcons=NONE″
;
```

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

The following example shows a successful completion of a **AUD-PATHNBR** command for ONNS path neighbor data audit operation by the NE with all data in sync:

```
AUD-PATHNBR:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2006-01-01 08:00:00
M  123456 COMPLD
   ″MismatchedPathIDs=NONE″
   ″NotReachedNeighborNodes=NONE″
;
```

The following example shows a successful completion of a **AUD-PATHBR** command for ONNS path neighbor data audit operation by the NE with some data out of sync:

```
AUD-PATHNBR:LT-ONNS-789:snn:123456;
```

```
   LT-ONNS-789 2006-01-01 08:00:00
M  123456 COMPLD
   ″MismatchedPathIDs=1-1-4-4-1,-X-,1-2-1-3-1,sts12,29.13.18.23-
      00000,29.16.29.111,29.14.196.203-32165
   :1-1-7-4-2,-X-,1-1-3-5-1,sts48,,29.13.18.23,29.14.196.109-12164″
   ″NotReachedNeighborNodes=29.13.18.24,29.18.24.30″
;
```

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

Error responses

Response specific to AUD:

Error code: `E650 – AAIP – Status, Audit Already In Progress`

Related TL1 messages

- “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698)
- “ED-SYNCDATA: Synchronization of ONNS data” (p. 3-1198)
- “RTRV-AUDLOG: Retrieve audit result log for ONNS connection data” (p. 3-2263)

Possible issues when widely using unidirectional ONNS connections

When in an ONNS network unidirectional ONNS connections are widely used, the following misbehavior may occur:

1. Local audit erroneously reports extra NN cross-connections: **AUD-LOCAL** and **RTRV-AUDLOG** may erroneously report extra NN cross-connections, if two unidirectional ONNS connections of differing bandwidth are traversing a node in opposite directions, but with connecting the same timeslots, for example, one connection from timeslot A to timeslot B and the other one from timeslot B to timeslot A.
2. Free bandwidth generally advertised on an INNI link, although no more bandwidth for bidirectional ONNS connections available: Setup of a bidirectional ONNS connection via an INNI link may fail, although for this link enough free bandwidth is generally advertised for example, via **RTRV-NTDR**. This is the case, if the corresponding link is nearly totally filled with a major part being unidirectional connections. The advertised free bandwidth may then refer to differing timeslots for both directions, thus not allowing bidirectional connections to be setup.

CANC-L1ROUTING

Overview

The command **`CANC-L1ROUTING`** is used to remove an ONNS enabled node as the proxy of its own domain in the routing hierarchy level 1. The command **`CANC-L1ROUTING`** can only be executed in maintenance mode.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command syntax: **`CANC-L1Routing:`***srcTid:srcAid:ctag;*

Output format

Normal sytem response format to CANC-L1ROUTING

If the CANC-L1ROUTING request accepted by the system, the following normal completion response format below is returned:

```
  srcTid date time
M  ctag COMPLD
;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:

YY-MM-DD, where

- YY=00-99,
- MM=01-12,
- DD=01-31

Default: current date

3. **`time`**
   Time. The time parameter is set to the time when the output message is generated.
   See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time
4. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.
5. **`completion code`**
   Standard TL1 completion code result.
   Type: Value set.
   Range:
   - COMPLD (default): for success
   - DENY: in case of errors

Example input/output

Input/Output example to disable a level 1 Routing Controller

The following example shows the successful completion of a CANC-L1ROUTING command to disable an ONNS node as level 1 Routing Controller:

```
CANC-L1ROUTING:LT-ONNS9:snn:113456;

   LT-ONNS9 2002-07-18 08:00:00
M  113456 COMPLD
```

;

Error responses

Validation rule: level 1 Routing controller does not exist

If there is no level 1 Routing Controller provisioned on the addressed node, the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Neighboring routing controller still exists

If the neighboring routing controller still exists (via DLT-RC-NEIGHBOR), then the command is denied.

Error code: `SNVS – Status, Not in Valid State, RC neighbor still exists`

Validation rule: Check of the maintenance condition

The system must be in maintenance condition (MCOND=YES) to remove an ONNS enabled node as the proxy of its own domain in the routing hierarchy level 1.

Error code: `SNVS – Status, Not in Valid State`

Related TL1 messages

- "ACT-L1ROUTING" (p. 3-15)
- "REPT L1ROUTING notification" (p. 3-2133)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)

CANC-L1ROUTING notification

The **`CANC-L1ROUTING`** notification message reports if an ONNS node is disabled as level 1 Routing Controller. Disabling is caused by a user's TL1 command.

Notification messages resulting from a TL1 command that were caused by the user's own requested changes

If an ONNS node is enabled as level 1 Routing Controller as a result of a TL1 CANC-L1ROUTING command, then a REPT DBCHG message is transmitted to all users provisioned to receive REPT DBCHG messages, including the user that issued the command. Therefore, a management OS must expect to receive messages for changes it caused as well as those made by a CIT user.

Output format

The REPT DBCHG message shall use the following format:

```
   sid date time
A  atag REPT DBCHG
   ″TIME=chgtm,DATE=chgdat:ccb:[aid]:[com_block]:[spec_block]″
     ...
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

1. **`atag`** – Automatic Tag
   The *atag* is used for message sequencing. The *atag* is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Type: decimal number.
   Range: 0 to 999.
   Default: 0
2. **`chgtm`** – Change Time
   Time when the change is occurred.
   Type: value set.
   Range: HH-MM-SS.
3. **`chgdat`** – Change Date
   Date when the change occurred
   Type: Value set.
   Range: YY-MM-DD

4. **`ccb`** – Command Code Block
   This field will be set to the TL1 command that causes that database change. The *com_block* and *spec_block* fields will be set according to the parameters that changed.
   Type: Alphanumeric value set verb-modifier[-modifier].
5. **`aid`** – Access Identifier
   AID of the entity for which the change is reported
   Type: AID.
   Range: As defined by the *aid* input parameter of the corresponding command, defined by *ccb*.
6. **`com_block`** – Common Block
   This is a position-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter values are included in this block in the same order as the corresponding TL1 command.
   As exception to this rule are some commands in the protection fragment, where the aliases *rid* and *ppgname* are not reported in the **`REPT DBCHG`** notification.
   If the database change is the result of a TL1 command, then the parameter *ctag* from the TL1 command is not included in the *com_block*.
   Type: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
7. **`spec_block`** – Specific Block
   This is a name-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter names and values are included in this block. Parameters within the specific block are positionally independent and are specified using a name-defined construct of *parameter* = value in a comma-separated list. The parameter names are the same as those for the corresponding TL1 command.
   Type: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.

Detailed behavior description

The **`REPT DBCHG`** message reports any change in the system database. Changes can be caused by a TL1 user or by an internally-generated state change.

Changes in user provisionable attributes are reported as the equivalent TL1 command that would cause that change.

Changes in non-user provisionable attributes are reported using only the AID and the block parameters, i.e. leaving the *ccb* parameter empty.

System initialization

Any time a command is entered that causes a system reset, a **REPT DBCHG** message is transmitted before the system resets.

AID ranges

If a database change occurs as a result of a TL1 command in which an AID range is entered, the system responds separate **REPT DBCHG** messages for each affected AID.

Messages caused by own changes

If the database change occurs as a result of a TL1 command, a **REPT DBCHG** message is transmitted to users provisioned to receive **REPT DBCHG** messages, including the user which issued the command.

Example input/output

The following is an example of the **REPT DBCHG** command:

```
   UNITE-NE 00-11-08 08:00:00
A  001 REPT DBCHG
   ″TIME=07-59-59,DATE=00-11-08: ENT-EQPT:1-1-#-#-05::nvmname=OP2G5,nvmqual=13SR4″
;
```

The following example shows the output of a REPT L1ROUTING notification caused by a CANC-L1ROUTING TL1 command:

```
   LT-ONNS9 2002-07-18 08:00:00
A  113456 REPT DBCHG
   ″TIME=07-59-59,DATE=2002-08-01:CANC-L1ROUTING″;
```

References

GR-199-CORE

CANC-USER: Cancel user

The **CANC-USER** command can be initiated to terminate the login session of the user with the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): Any

Priority: 1

Abortable: No

Input format

Command syntax: **CANC-USER**:*tid*:[*uid*]:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **uid** – User Identifier
   This is the user identifier that was used in the **ACT-USER** command that initiated this session. This is an optional parameter; if it is not specified, it defaults to the user identifier of this session.
   The uid is not really needed in this command, because always the own session will be canceled. But this parameter is in because of telcordia requirements.
   This is the unique user login identifier for which the this command is intended.
   Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed.
   Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 5 – 10 characters. The *uid* ALL is not allowed in any combination of upper or lower cases.
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

If the logout request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

When the NE receives a CANC-USER command, the login session on the NE will be terminated provided that the user is currently active on the NE. Only the users own session can be terminated.

The **`CANC-USER`** command does not generate a **`REPT DBCHG`** message.

Example input/output

The following example shows a successful command completion and login session termination:

```
CANC-USER:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: not the own uid specified

Only the own *uid* must be specified. There is not so.

Error code: `IDNC – Input, Data, Not Consistent, not own uid`

Related TL1 messages

- "ACT-USER: Activate user" (p. 3-24)
- "CANC-USER-SECU: Cancel user security" (p. 3-62)
- "DLT-USER-SECU: Delete user security" (p. 3-272)
- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "ENT-USER-SECU: Enter user security" (p. 3-1970)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

CANC-USER-SECU: Cancel user security

The execution of this command on a NE by an security administrator terminates all login sessions of a particular user or of all non-superusers/non-MEMADMIN-users within the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No

Input format

Command syntax: **`CANC-USER-SECU`**:*tid:[uid]:ctag;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`uid`** – User Identifier
   This is the *uid* of the user whose sessions are being terminated. If an security administrator wants to force log out another user, then the security administrator could do so by specifying that *uid*. An privileged user could log out any other user including super-users and users of *uidclass* MEMADMIN.
   If an security administrator wants to force log out all the users that are currently logged in, then the security administrator could do so by omitting the *uid*. Logging out all users by omitting the *uid* does not include the super-users or MEMADMIN-users logins.
3. **`ctag`** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

If the logout request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **CANC-USER-SECU** command can be only executed by the security administrator (UPC S5). She/he can terminate the sessions of other users by supplying his/her *uid*. If a user has established more than one sessions then all sessions will be terminated. The own session will not be affected this command.

The **CANC-USER-SECU** command does not generate a **REPT DBCHG** message.

Example input/output

The following example shows a successful command completion and login session termination:

```
CANC-USER-SECU:LT-UNITE-123:nbguser1:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: uid not in NE data base

The *uid* is not contained in the NE’s data base.

Error code: `IDNV – Input, Data Not Valid, uid not in NE data base`

Validation rule: user with uid not logged in

A user with the specified *uid* is not currently logged in.

Error code: `SRQN – Status, invalid request, uid not logged in`

Validation rule: uid must not be the own uid

If the own uid matches *uid* the command should be denied.

Error code: `IDNV – Input, Data Not Valid, uid must not be the own uid`

Related TL1 messages

- “ACT-USER: Activate user” (p. 3-24)
- “CANC-USER: Cancel user” (p. 3-60)
- “DLT-USER-SECU: Delete user security” (p. 3-272)
- “ED-NE-SECU: Edit NE security” (p. 3-446)
- “ED-USER-SECU: Edit user security” (p. 3-1328)
- “ENT-USER-SECU: Enter user security” (p. 3-1970)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

CHG-ACCMD: Change access mode

General

The **CHG-ACCMD** command is used to change the test access mode of a test session in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **CHG-ACCMD**-*modifier:tid:tatrib1:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the connect test access command is to act. Type: Alphanumeric value set.
   Range:
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **tatrib1** – primary Test Access tributary
   AID of the primary Test Access tributary. The keyword **all** is not allowed on any level.
   Type: AID
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **tstmde** – Test Access Mode
   Type: Alphanumeric value set.

Range: Allowed values:

- MONE – Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.
- MONF- Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.
- MONEF – Monitor E-end and F-end mode of test access. This test access mode monitors (E end and F end) both directions of a two-way cross-connection.
- SPLTE – Split access E-end mode of test access. This test access mode splits the connection between the E end and F end of a one-way cross-connection or a leg of two-way cross-connection. Can also be used in an idle tributary.
- SPLTF – Split access F-end mode of test access. This test access mode splits the E-end and F-end connection of a two-way cross-connection to allow testing at the F-end.
- SPLTEF – Split E-end and F-end mode of test access. This test access mode splits both directions of a two-way cross-connection to allow testing at both the E end and F end.

Output format

The NE returns the following normal completion response to a successful CHG-ACCMD command if there are no failure conditions to report:

```
   sid date time
M  ctag COMPLD
;
```

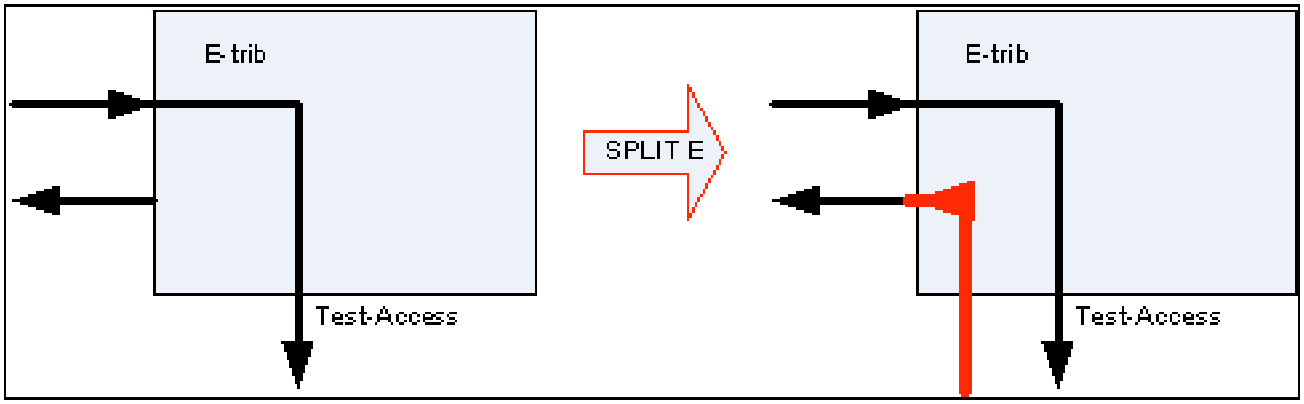
Detailed behavior description

The function of the **CHG-ACCMD** command is to change the test access mode without terminating the test access session.

The test access mode changes are restricted to test modes belonging to the same group. (See the definition of the groups in the general description.) It is not allowed to change the mode from single to dual test tributary modes or from dual to single test tributary modes.

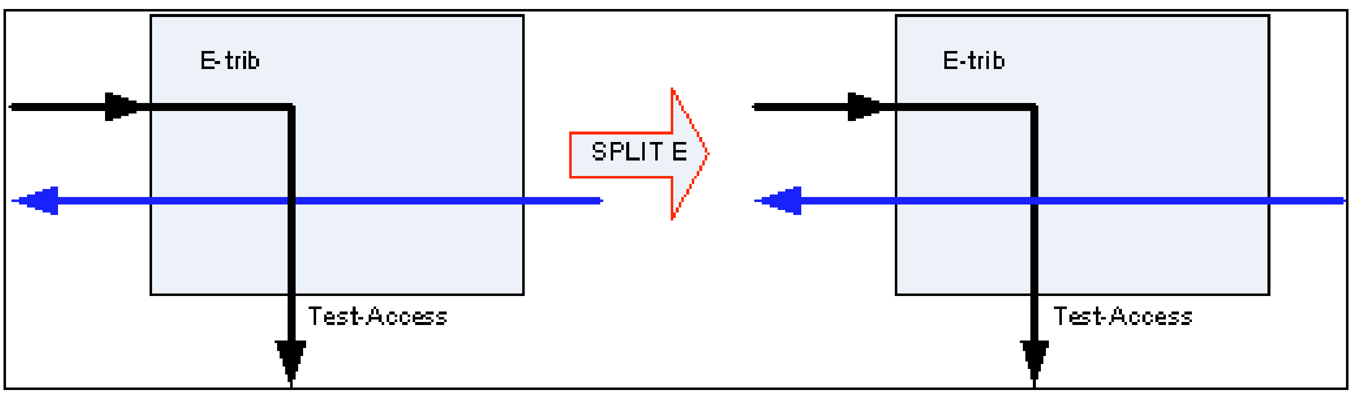
Idle tributary 1

If the “Split E” (SPLTE) mode is applied on an idle tributary which is also not the destination of a cross-connection- then the test access signal is connected from the test access tributary to the E-tributary.



Idle tributary 2

If the “Split E” (SPLTE) mode is applied on an idle tributary which is the destination of a cross-connection, then the test access signal is not connected from the test access tributary to the E-tributary. There is no cross-connection to be created.



Example input/output

```
CHG-ACCMD-STS1:LT-UNITE:1-1-#-#-6-1-1:123456:::TSTMDE=SPLTA;
   LT-UNITE 02-10-01 19:17:53
M  123456 COMPLD
;
```

For lower-order test access sessions:

```
CHG-ACCMD-LOVT1:LT-UNITE:1-1-#-#-6-1-1-28:123456:::TSTMDE=SPLTA;
   LT-UNITE 02-10-01 19:17:53
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **CHG-ACCMD** command.

Validation rule: No test access session on the test tributary

If the NE receives a **CHG-ACCMD** command when there is no test access session set up on the test tributary specified, the following error response is returned:

Error code:
`SNVS – Status, Not in Valid State, no test access session on the test tributary`

Validation rule: Invalid test access mode change request

If the NE receives a **CHG-ACCMD** command with a test access mode which is not a member of the same group, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, invalid test access mode`

Validation rule: Underlying cross connection is not two-way

If the network element receives a **CHG-ACCMD** command which changes the tributary under test from tributary E to tributary F and the underlying cross-connection is not two-way, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, invalid test access mode`

Validation rule: Underlying cross connection belongs to BLSR/MS-SPRing

If the network element receives a **CHG-ACCMD** command which requests any kind of Split mode (SPLT) and the underlying cross-connection is a HO BLSR/MSSPRing intra-ring through cross-connection, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, invalid test access mode`

Validation rule: Inconsistent tributary rate

If the NE receives an **CHG-ACCMD** command which does not match the test access connection rate, the following error response is returned:

Error code:
`IPNC – Input, Parameter Not Consistent, Rate inconsistent with crossconnection rate`

Validation rule: Not Supported for Low-Order Test Access Rate

If the network element receives an **CHG-ACCMD** command with modifier LOVT1 the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, Command not supported for test access rate LOVT1.5`

Note: Wrong error code may occure:

When it is tried to change the test mode of a LO test access session with the command **CHG-ACCMD**, then the command is denied witht the wrong error code:

```
SROF – Status, Requested Operation Failed
```

Related TL1 messages

- "CONN-TACC: Connect test access" (p. 3-94)
- "RTRV-TACC: Retrieve test access" (p. 3-2974)
- "DISC-TACC: Disconnect test access" (p. 3-129)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)

CHG-ACCMD notification

Output format

The **`CHG-ACCMD`** notification has the following specific syntax:
`CHG-ACCMD`-*modifier:tatrib1::spec_block;*

Output parameters

Command parameters

1. **`modifier`**
   This parameter indicates the cross-connection rate on which the connect test access command is to act. Type: Alphanumeric value set.
   Range:
   - SDH modifier: LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **`tatrib1`** – primary Test Access tributary
   The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.

Spec block parameters

1. **`eomode`** – output mode of the E-tributary
   This parameter is only present if the test session is created on a two-way cross-connection. The parameter is omitted for test sessions on one-way cross-connections or on idle E-tributaries.
   Type: Alphanumeric value set.
   Range:
   - NORM: this is the normal output i.e. the signal from the F-tributary is passed through to the E-tributary
   - AIS: this mode outputs an AIS signal.

   Default:
   - NORM, if the referring Ttest mode does not disconnect the output of the original cross-connection to this tributary
   - AIS, in all other case.
2. **`fomode`** – output mode of the F-tributary
   This parameter is only present if the test session is created on a one-way or two-way cross-connection. The parameter is omitted for test sessions on idle E-tributaries.
   Type: Alphanumeric value set.

Range:

- NORM: this is the normal output i.e. the signal from the E-tributary is passed through to the F-tributary
- AIS: this mode outputs an AIS signal.

Default:

- NORM, if the referring test mode does not disconnect the output of the original cross-connection to this tributary
- AIS, in all other cases.

3. **`tstmde`** – Test Access Mode
   Type: Alphanumeric value set.
   Range: Allowed values:
   - MONE – Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.
   - MONF- Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.
   - MONEF – Monitor E-end and F-end mode of test access. This test access mode monitors (E end and F end) both directions of a two-way cross-connection.
   - SPLTE – Split access E-end mode of test access. This test access mode splits the connection between the E end and F end of a one-way cross-connection or a leg of two-way cross-connection. Can also be used in an idle tributary.
   - SPLTF – Split access F-end mode of test access. This test access mode splits the E-end and F-end connection of a two-way cross-connection to allow testing at the F-end.
   - SPLTEF – Split E-end and F-end mode of test access. This test access mode splits both directions of a two-way cross-connection to allow testing at both the E end and F end.

Example input/output

```
CHG-ACCMD-STS1:LT-UNITE:1-1-#-#-6-1-1:123456:::TSTMDE=SPLTA;

   LT-UNITE 02-10-01 19:17:53
M  123456 COMPLD
;

   LT-UNITE 02-10-01 19:18:22
A  001 REPT DBCHG
   ″TIME=19-18-05,DATE=02-10-01:CHG-ACCMD-STS1:1-1-#-#-4-1-1::TSTMDE=SPLTA:″
;
```

For lower-order test access sessions:

```
CHG-ACCMD-LOVT1:LT-UNITE:1-1-#-#-6-1-1-28:123456:::TSTMDE=SPLTA;
```

```
   LT-UNITE 02-10-01 19:17:53
M  123456 COMPLD
;

   LT-UNITE 02-10-01 19:18:22
A  001 REPT DBCHG
   ”TIME=19-18-05,DATE=02-10-01:CHG-ACCMD-LOVT1:1-1-#-#-4-1-1-28::\
    TSTMDE=SPLTA:”
;
```

CNVT-CCT: Convert cross-connection topology

General

The **CNVT-CCT**-modifier command can be initiated by a user to convert between a 1-way point-to-point cross-connection and a path-protected cross-connection.

For a 1-way Point-to-Point to a path-protected conversion, the existing point-to-point cross-connection leg is converted into a single atomic path-protected cross-connection between the logical input AID of the existing point-to-point cross-connection, the specified protection logical input AID and logical output AID of the existing point-to-point cross-connection. The leg comprised of the AIDs of the existing point-to-point cross-connection is designated as the path protection group working leg at the time the cross-connection is converted.

If the command is applied to a 2-way cross-connection it will be split up automatically in two 1-way cross-connections. The command will be executed on the specified in_aid to out_aid direction, the reverse direction will remain unprotected. Refer also to chapter Detailed behavior for more.

For a path-protected to a 1-way point-to-point conversion, it can be selected, which logical input AID of the existing legs (working leg or the protection leg) of the path-protected cross-connection shall stay as logical input AID of the created point-to-point cross-connection. This can be done by specifying the surviving leg of the existing path protection group as working leg at the time the cross-connection is converted.

The tributaries involved do not need to be free of alarms/failure conditions – no check in advance will be made. The cross-connection will thus be established also in case of AIS-P, LOP-P, UNEQ etc. as is the case with a normal ENT-CRS command as well.

The *modifier* in the **CNVT-CCT** command indicates the cross-connection rate of the tributary for which the command is applied to. The same rules as for ENT-CRS for path-protected cross-connections apply.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 3

Abortable: No

Input format

Command syntax: **CNVT-CCT**-*modifier*:*tid*:*in_aid,out_aid:ctag::[in_aid2]:spec_block;*

Input parameters

Command parameters

1. **`modifier`**
   This parameter indicates the cross-connection on which the **`CNVT-CCT`** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192.

   No use of the **`all`** value is allowed.
2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`in_aid`** – Input Access Identifier
   This parameter specifies
   - 1-way Point-to-Point to a Ppath-protected conversion: the source of the unprotected cross-connection respectively
   - Path-protected to a 1-way point-to-point conversion: the AID of the input leg which shall stay as source of the created unprotected cross-connection.

   Type: AID
   Range: GenTribAID or GenLOTribAID. No use of the *all* or *usall* or *vall* values are allowed.
   No use of the **`all`** value within the in_aid2 is allowed.
   VCGNumAID is excluded from the value range by validation rule.

4. **`out_aid`** – Output Access Identifier
   This parameter specifies the sink of the cross-connection. See definition of parameter *out_aid* in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
   Type: AID
   Range: GenTribAID or GenLOTribAID.No use of the *all* or *usall* or *vall* values within the out_aid is allowed.
5. **`ctag`** – Correlation Tag
   See definition of *ctag* in “RTRV-HDR: Retrieve header” (p. 3-2417).
6. **`in_aid2`** – Input2 Access Identifier
   2nd Input Access Identifier. For a Path-Protected to a 1-way Point-to-Point conversion, the parameter in_aid2 specifies the input leg which is dropped.
   For a 1-way Point-to-Point to a Path-Protected conversion, the parameter has to be omitted.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID
   Range: No use of the *all* or *usall* or *vall* values are allowed.
   Default: a valid value is needed on case of a path-protected cross-connection – no default is defined.

Spec block parameters

1. **`cct`** – CrossConnect Topology
   The parameter *cct* specifies the new cross-connection topology.
   Type: Alphanumeric value set.
   Range: 1WAY for one-way unprotected cross-connection; 1WAYPPROT for one-way path-protected cross-connection.
2. **`htmr`** – Hold-Off Time
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine. Type: numeric string [ms].
   Type: numeric string [ms]
   Range: For high order path protection groups, the range is: 0 – 99 ms in 1 ms steps and 100 – 10000 ms in100 ms steps. For low order path protection groups, the range is: 0, 25, 50, 75 ms (thus in steps of 25 ms) and 100 – 10000 ms in 100 ms steps.
3. **`newinaid`** – New Input Access Identifier
   This parameter specifies the source aid of the new leg when converting from a point-to-point to a path-protected cross-connect. For removing the path-protection this parameter does not apply and is omitted.
   Type: AID.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID
   No use of the **`all`** or **`usall`** values are allowed.
4. **`newloca`** – New Location-a
   This parameter specifies the new source TID if the *newinaid* is in a BLSR protection group. It specifies that add node which is related to the protection leg input. *newloca* will be stored as the *loca2* parameter.
   The parameter is mandatory, if *in_aid2* is at a BLSR/MS-SPRing protection group

with Automatic Squelch Map Calculation disabled.
The parameter *loca2* (location-a-2) specifies the TID associated with *in_aid2* of the protection leg of a path-protected cross-connection. For the syntax, range and default definition refer to *loca* in “ENT-CRS: Enter cross-connection” (p. 3-1464).
Note that the *loca-2* parameter is omitted if no *in_aid2* has been specified or for non-BLSR cross-connection.
Additional information for this parameter is provided in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
Type: TID
Range: 1 to 20 characters

5. **occt** – Original CrossConnect Topology
The parameter cct specifies the original cross-connection topology.
Type: Alphanumeric value set.
Range: 1WAY for one-way unprotected cross-connection. 1WAYPPROT for one-way path-protected cross-connection.
6. **ppgname** – Path Protection Group Name
This parameter specifies the name of the path protection group to be established. For removing the path-protection this parameter does not apply and is omitted. See definition of *ppgname* parameter in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
7. **rme** – Protection Reversion
The parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group. The value ENABLE is used if the protection group operates in revertive mode. The value DISABLE is used if the protection group operates in non-revertive mode.
Type: Alphanumeric value set Range: DISABLE in case of protype = 1+1; ENABLE in case of protype = 1xNOPT.
Default: DISABLE, in case of protype = 1+1; ENABLE, in case of protype = 1xNOPT (i.e. protection type 1:1) The protocol = OPTM can only be chosen in case of protype = 1+1, thus default is DISABLE then.
8. **snctype** – SNC Protection Type
This parameter specifies the type of the path protection w.r.t. switch criteria used by the protection state machine. The value SNCI (Inherent) is used if server signal fail is used as switch criteria. The value SNCN (Non-Instrusive) is used if server signal fail and defects supervised by a non-intrusive path monitor are used as switch criteria. The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
Type: Alphanumeric value set.
Range: SNCI, SNCN, NA.
9. **wtr** – Wait To Restore Time
This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
Type: numeric string [minute].
Range: 0 – 60.

10. **xcappl** – cross-connection application
    This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies. The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.
    The numeric values are assigned to appropriate text strings (see Range below) and the manager should display these value in case of a "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).

    As a possible assignment proposal the following may serve:

    - 0) Unknown
    - 1) 1-Way Point-to-Point
    - 2) 1-Way Path-Protected
    - 3) 1-Way Adjunct Path-Protected
    - 4) 1-Way Conversion Cross-Connection
    - 10) 2-Way Point-to-Point
    - 11) 2-Way Conversion Cross-Connection
    - 20) UPSR (or SNCP Ring) Add, Drop
    - 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
    - 22) UPSR (or SNCP) Drop, 1-Way Broadcast
    - 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
    - 30) Logical Ring (or SNCP) Add, Drop
    - 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
    - 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
    - 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
    - 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)
    - 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
    - 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

    Type: numerical value set.
    Range: {0, 1, 2,…, 255}.
    Default: The default value is 0.
11. **xcnum** – cross-connection number
    This parameter is a multiple digit number, identifying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. described in ″Example input/output″ section of the **RTRV-CRS** command like UPSR,

DRI etc.).

Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in ring interworking scenarios).

The following format for the twelve digits is recommended:

- two digits for the bay
- one digit for the shelf
- two digits for the slot
- two digits for the port
- three digits for the high-order tributary
- two digits for the low-order tributary.

If a high-order tributary is addressed then the last two digits are set to 00. The digits values pertain to the *out_aid*.

Type: numeric value set (12 digit).

Range: 000000000000 … 999999999999.

Default: The default value is 000000000000.

Output format

If the **CNVT-CCT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Notification of cross-connection related changes

Because the CNVT-CCT can be understood as a sequence consisting of a DLT-CRS operation followed by an ENT-CRS operation, a (separate) CNVT-CCT notification does not exist. Instead a DLT-CRS notification is sent, followed by an ENT-CRS notification.

Refer also to the section detailed behavior of DLT-CRS and ENT-CRS.

Keep in mind that the attribute *in_aid2* is a part of the key / identifier for the path-protection group, and therefore the above described deletion and creation sequence is a proper approach in order to notify the performed DB changes.

The ENT-CRS notification includes already information about the changed *repleg* status.

Notification of protection group related changes

The ENT-CRS operation affects subsequently also the protection area, i.e.in case of entering a path-protected cross-connection a path protection group will be established. Thus two independent notifications are generated: ENT-CRS notification and a ENT-PROTN-GRP notification.

The DLT-CRS operation affects subsequently also the protection area, i.e. in case of removing a path-protected cross-connection a path protection group will be deleted. Thus two independent notifications are generated: DLT-CRS notification and a DLT-PROTN-GRP notification.

Transaction behavior

For the **`CNVT-CCT`** command the validation is performed before the COMPLD is returned. This means it has to be checked if the DLT-CRS as well as the following ENT-CRS can be performed successfully, before the entire command is executed. With this approach it is guaranteed that the command either is rejected or succeeds as a whole by executing the deletion- AND the entering-operation.

For the detailed validation rules refer to the chapter ’error responses’ below.

Updated affected entities: 1way, 2way cross-connection, path-protected cross-connection

The establishing of a path-protection group on a possibly existing 2-way cross-connection leads to a conversion to a 1-way path-protected cross-connection plus a separate 1-way cross-connection for the return direction. A similar split-up happens when deleting one direction of a 2-way cross-connection. Therefore refer to section ″Detailed behavior description″ of the DLT-CRS notification.

After deletion of a path-protection group the remaining 1-way cross-connection may have an equivalent return leg. Then both will be merged in a 2-way cross-connection. A similar merge happens when a cross- connection is established to which already an equivalent return leg is existing. Therefore refer to section ″Detailed behavior description″ of the ENT-CRS notification.

Auto-substructuring and auto-desubstructuring of HO tributaries

If a leg of a low-order cross-connection is being created or deleted due to a topology conversion command then:

- in the case of cross-connection leg creation: the related HO tributary gets properly auto-substructured if the HO tributary is not already substructured. HO PTFs which support only LO but no HO connectivity are not explicitly substructured
- in the case of cross-connection leg deletion: the related HO tributary gets properly auto-desubstructured if the HO tributary has been auto-substructured (i.e. substr = YES). HO PTFs which support only LO but no HO connectivity are not desubstructured. Cross-connection leg creation occurs during conversion to 1WAYPROT, deletion during conversion to 1WAY.

Auto-reconfiguration of the PTF substructure (SDH only)

If a leg of a low-order SDH cross-connection is being created due to a topology change (to 1WAYPROT) such that a new third of substructured HO tributary gets involved which does not match the rate, then this third of the substructure gets adapted as specified in section"Low-order cross-connection specific behavior - Auto-reconfiguration of the PTF substructure (SDH only)" (p. 3-1478) of "ENT-CRS: Enter cross-connection" (p. 3-1464). It does not matter whether the new third is part of the same HO tributary or not. Pre-condition for this auto-adaption is still that all affected LO tributaries are not yet involved in a cross-connection before the topology conversion command is executed.

Mandatory, optional and not allowed parameters

- for converting to a higher-order non-pathprotected cross-connection
  - the following parameters must be present: *in_aid2*, *cct*, *occt*
  - the following parameters must not be present: *htmr*, *newloca*, *rme*, *snctype*, *wtr*
  - the following parameters may be present: *newinaid*, *ppgname*
- for converting to a lower-order non-pathprotected cross-connection
  - the following parameters must be present: *in_aid2*, *cct*, *occt*
  - the following parameters must not be present: *htmr*, *newloca*, *rme*, *snctype*, *wtr*, *pgname*, *newinaid*
- for converting to a higher-order pathprotected cross-connection with *newinaid* at a port that is **not** in a BLSR/MS-SPRing protection group,
  - the following parameters must be present: *cct*, *htmr*, *newinaid*, *occt*, *rme*, *snctype*, *wtr*
  - the following parameter must not be present: *in_aid2*, *newloca*
  - the following parameter may be present: *ppgname*

- for converting to a higher-order pathprotected cross-connection with *newinaid* at a port that is part of a BLSR/MS-SPRing protection group,
  - the following parameters must be present: *cct*, *htmr*, *newinaid*, *occt*, *rme*, *snctype*, *wtr*, *newloca* (only mandatory if Automatic Squelch Map Calculation is disabled)
  - the following parameter must not be present: *in_aid2*
  - the following parameter may be present: *ppgname*
- for converting to a low-order pathprotected cross-connection without nesting BLSR/MS-SPRing
  - the following parameters must be present: *cct*, *htmr*, *newinaid*, *occt*, *rme*, *snctype*, *wtr*
  - the following parameters must not be present: *in_aid2*, *newloca*, *ppgname*
- for converting to a low-order pathprotected cross-connection with nesting BLSR/MS-SPRing
  - the following parameters must be present: *cct*, *htmr*, *newinaid*, *newloca*, *occt*, *rme*, *snctype*, *wtr*
  - the following parameters must not be present: *in_aid2*, *ppgname*

Example input/output

Example 1:

Suppose the following existing non-BLSR 1way vc3-cross-connection

1-1-#-#-1-1-3 -> 1-1-#-#-5-1-17

is protected by the new leg

1-1-#-#-2-1-3 -> 1-1-#-#-5-1-17

also being a non-BLSR cross-connection. This would be accomplished via:

```
CNVT-CCT-VC3:UNITE-NE:1-1-#-#-1-1-3,1-1-#-#-5-1-17:123456:::
occt=1WAY,cct=1WAYPPROT,newinaid=1-1-#-#-2-1-3,
ppgname=”group018”,rme=ENABLE,snctype=SNCN,htmr=100,wtr=5;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 2:

Suppose the following existing path-protected vc3-cross-connection:

1-1-#-#-1-1-3 -> 1-1-#-#-5-1-17 working leg

1-1-#-#-2-1-3 -> 1-1-#-#-5-1-17 protection leg is changed into an unprotected cross-connection via the working leg. This would be accomplished via:

```
CNVT-CCT-VC3:UNITE-NE:1-1-#-#-1-1-3,1-1-#-#-5-1-17:123456::1-1-#-#-2-1-
      3:occt=1WAYPPROT,cct=1WAY;

   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 3:

Suppose the following existing path-protected vc3-cross-connection:

1-1-#-#-1-1-3 -> 1-1-#-#-5-1-17 working leg

1-1-#-#-2-1-3 -> 1-1-#-#-5-1-17 protection leg

is changed into an unprotected cross-connection via the protection leg. This would be accomplished via:

```
CNVT-CCT-VC3:UNITE-NE:1-1-#-#-2-1-3,1-1-#-#-5-1-17:123456::1-1-#-#-1-1-
      3:occt=1WAYPPROT,cct=1WAY;

   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example3 Notification behavior:

First the deletions of the path protected cross-connection and the protection group are notified, then the creation of the 1WAY cross-connection is notified with the previous protection leg as input tributary:

DLT-CRS-VC3:1-1-#-#-1-1-3,1-1-#-#-5-1-17:1WAYPPROT,1-1-#-#-2-1-3;

DLT-PROTN-GRP:1-1-#-#-5-1-17:PATHDRI;

ENT-CRS-VC3::1-1-#-#-2-1-3,1-1-#-#-5-1-17:1WAY, …

Example 4:

The following lower order 1way vc3-cross-connection:

1-1-#-#-1-2-1-1 -> 1-1-#-#-5-2-16-22 is protected by the new leg

1-1-#-#-2-2-4-1 -> 1-1-#-#-5-2-16

Example 43 Notification behavior:

If the cross-connection conversion at LOVC3 rate causes the auto-substructuring of a HO NIM entity then first the creation of the default substructure (6312) will be reported. After then, an auto-reconfiguration of the PTF substrcuture will take place in order to adapt the affected third to LOVC3. This causes the following notifications:

ED-VC3:1-1-#-#-2-2-4::substr=YES;

ENT-PTF:1-1-#-#-2-2-4::inputsig=6312,unequiposig=6312,…

ENT-LONIM:1-1-#-#-2-2-4-1&&-63::pmlpfeses=600,pmlpneses=600,…

ED-PTF:1-1-#-#-2-2-4::inputsig=13-4212;

ED-LONIM:1-1-#-#-2-2-4-1::pmlpfeses=2400,pmlpneses=2400;

The following notifications are for cross-connection deletion/creation caused by the **CNVT-CCT** command:

DLT-CRS-LOVC3::1-1-#-#-1-2-1-1,1-1-#-#-5-2-16-22:::1WAY;

ENT-CRS-LOVC3::1-1-#-#-1-2-1-1,1-1-#-#-5-2-16-22:::1WAYPROT,1-1-#-#-2-2-4-1:

repleg=1WAYPSW …

ENT-CRS-LOVC3::1-1-#-#-2-2-4-1,1-1-#-#-5-2-16-22:::1WAYPROT,1-1-#-#-1-2-1-1:

repleg=1WAYPSP …

and for protection group creation: ENT-PROTN-GRP:1-1-#-#-5-2-16-22::PATHDRI:protn=1-1-#-#-2-2-4-1,wkg=1-1-#-#-1-2-1-1, …

Error responses

The **CNVT-CCT** procedure can be conceptually understood as a sequence consisting of a **DLT-CRS** operation followed by an **ENT-CRS** operation. Therefore some of the validation rules of **DLT-CRS** and **ENT-CRS** shall apply also here. This means also that if one of the below listed **CNVT-CCT** validation rules would fail to meet a criteria the same or at least a similar command denial as specified in “DLT-CRS: Delete cross-connection” (p. 3-145) and “ENT-CRS: Enter cross-connection” (p. 3-1464) section will be the consequence.

Validation rules relevant for path-protection which differ:

- check of provisioned topology parameters against existing atomic topology,
- check of existence of an cross-connection via the (future respectively current) working leg.

Adaptation of validation rules

In general the validation rules of **DLT-CRS** (as stated in “DLT-CRS: Delete cross-connection” (p. 3-145)) and **ENT-CRS** (as stated in “ENT-CRS: Enter cross-connection” (p. 3-1464)) apply. However, if an atomic cross-connection will remain finally unchanged then for this part no validation rule must be performed. The full list of validation rules from the **DLT-CRS** / **ENT-CRS** area is given in the following requirements.

In general, the validation rules of **DLT-CRS** apply in case of removing a path-protection. However the check shall skip that the cross-connection of the working leg should have been removed before.

In general, the validation rules of **ENT-CRS** apply in case of establishing a path-protection. However the check shall allow that a cross-connection via the working leg has already been set up.

Validation rule: No HO cross-connections while LO cross-connections, substructure or conversion cross-connections exist

If the high-order tributaries specified in the high-order cross-connection command are substructured or LO cross-connections or conversion cross-connections exist at the HO tributaries, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: *newinaid* must not be involved in conversion cross-connections

If the parameter *newinaid* specifies

- a LO tributary which is used for a conversion cross-connection or
- a LO tributary which is related to a PTF where already another LO tributary is used for a conversion cross-connection, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: HO tributaries must exist

Cross-connection command processing shall verify the logical tributaries specified in the command do exist.

Error code:
`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Note the equivalence to the related ENT-CRS and DLT-CRS validation rules.

Validation rule: LO tributary must exist/be creatable

Cross-connection command processing shall verify that the low-order tributaries specified in the command do exist or can be properly created. For modifier LOVC3 a tributary boundary check is included, i.e. only the LO tributary numbers 1, 22, 43 are allowed.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

A connectable low-order entity may exist or not.

The existence is verified if:

- the LO NIM entity
- the HO PTF entity has a matching *inputsig* parameter (if used as input),
- the HO PTF entity has a matching *unequiposig* parameter (if used as output).

The possibility to create a related entity is verified if:

- the interface standard of the related port and the rate of the related HO NIM or HO PTF matches the cross-connect modifier (SDH/VC4 in case of LOVC3, LOVC12; SONET/STS1 in case of LOVT1)
- the HO NIM or HO PTF is not yet involved in any type of HO cross-connection*)
- the HO NIM is not yet substructured (parameter *substr* = NO)

The already cross-connected LO tributaries must exist. A newly added LO tributary may exist or may be creatable (thus not yet existing). Refer to "ENT-CRS: Enter cross-connection" (p. 3-1464) for examples.

*) this means:

- neither cross-connected, nor looped back, nor involved in a remote test access session.

Validation rule: Entity must be adaptable for LO cross-connections

Cross-connection command processing shall verify that the low-order tributary specified in newinaid can be properly created by adaptation of the existing substructure. For modifier LOVC3 a tributary boundary check is included, i.e. only the LO tributary numbers 1, 22, 43 are allowed.

Error code:

`IENE – Input, Entity Not Exists, Substructure cannot be adapted`

Validation rule: Entity (cross-connection) must exist

Cross-connection command processing shall verify that the specified cross-connection exists. This includes the check that the cross-connection rate matches the specified rate.

Error code:

`IENE – Input, Entity Not Exists, Specified crossconnection does not exist`

Note the equivalence to the related ENT-CRS and DLT-CRS validation rules.

Validation rule: LO cross-connection with specified topology must exist

Cross-connection command processing shall verify for lower order cross-connections that a lower order cross-connection with the specified topology (parameter occt)exists.

Error code:

`IENE – IENE – Input, Entity Not Exists, Specified crossconnection does not exist`

Note the equivalence to the related ENT-CRS and DLT-CRS validation rules.

Validation rule: Topology conversion command must be consistent

The spec block parameters *occt* and *cct* must be both provided and different.

Error code:

`IDNC – Input, Data, Not Consistent, no allowed topology conversion`

Validation rule: Only those parameter present which are needed

Cross-connection command processing shall verify according to the given context only those parameter are present which are needed for that context.

This means that

- for converting to a higher-order non path-protected cross-connection the following parameter must not be present:
  htmr, *newloca*, *rme*, *snctype*, *wtr*
- for converting to a lower-order non path-protected cross-connection the following parameter may be present:
  htmr, *newloca*, *rme*, *snctype*, *wtr*, *ppgname*, *newinaid*
- for converting to a higher-order path-protected cross-connection with *newinaid* at a port that is **not** in a BLSR/MS-SPRing protection group, the following parameters must not be present:
  in_aid2, *newloca*
- for converting to a higher-order path-protected cross-connection with *newinaid* at a port that is part of a BLSR/MS-SPRing protection group, the following parameters must not be present:
  in_aid2
- for converting to a lower-order path-protected cross-connection **without** nesting BLSR/MS-SPRing, the following parameters must not be present:
  in_aid2, *newloca*, *ppgname*
- for converting to a lower-order path-protected cross-connection **with** nesting BLSR/MS-SPRing, the following parameters must not be present:
  in_aid2, *ppgname*

Error code:

`SNVS – Status, Not in Valid State, Parameter(s) present which are not allowed`

CNVT-CCT: *ppgname* usage:

It is not possible for CNVT-CCT to identify the high-order path-protection group by the *ppgname* only. It has been excluded here. Reason: at least for a CNVT-CCT command with the order to create a protection group this is not sufficient to restrict it to a single cross-connection, as the *ppgname* may not be uniquely defined and therefore this might result in actions not desired by the operator.

Validation rule: New tributary rate validity

If converting to a path-protected cross-connection then the cross-connection command processing shall verify that the specified cross-connection rate is equivalent to the provisioned signal rate of the new input tributary identified by newinaid, as determined by the tributary input signal rate parameter.

Error code:

`IDNC – Input, Data, Not Consistent, Specified new Tributary does not support the indicated rate`

Note that for HO cross-connections it does not matter if the tributary signal rate is given for the SONET- (STS1/STS3/…) or the SDH-interface standard (VC3/VC4).

As an example: It is possible to enter a cross-connection via ENT-CRS-VC4… between two tributaries related to tributary signal rates given by VC4/VC4 or by VC4/STS3 or by STS3/STS3; Thus the inputsig/unequiposig-values for SDH (3, 4, 4c, 16c, 64c) are identical to those of SONET (1,3,12,48,192) seen from a cross-connection perspective.

For simplicity also the full validation rule of ENT-CRS may be executed.

For LO cross-connections, this validation is covered by the check, whether the LO substructure is adaptable to the new input tributary rate.

Validation rule: path-protected cross-connection has unique input AIDs

If converting to a path-protected cross-connection then the cross-connection command processing shall verify that for path-protected cross-connections the new input tributary identified by newinaid is different from *in_aid*.

Error code:

`IDNV – Input, Data Not Vaild, Both input tributaries are identical`

Note the equivalence to the related ENT-CRS validation rule.

Validation rule: Restrictions for conversion to LO path protection

If a low-order 1WAY cross-connection is converted to a path protected cross-connection, and the input tributary of the 1WAY cross-connection (in_aid) is already used as input tributary for another, already existing path protection, then the conversion is only allowed if: the specified input tributary (in_aid) of the 1WAY cross-connection is the working input tributary of the existing path protection AND the specified new input tributary (newinaid) is the protection input tributary of the existing path protection.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: Restrictions for LOXC cross-connection sources

If converting to a pathprotected cross-connection then the cross-connection command processing shall verify that the new input tributary identified by newinaid shall not reside: on the OPT2G5 pack, on the EP51 pack (DS3 and EC1 interfaces), on the GE1 pack, on the GE10PL1 pack, on the ES64U pack, at a port of any pack not being configured with portClass as TRADITIONAL or EDGE.

Error code:

`IDNC – Input, Data Not Consistent, Specified Tributary cannot be used for LOXC connections`

Validation rule: Spare capacity for HO cross-connection legs on LOXC

If converting to a pathprotected cross-connection then the cross-connection command processing shall deny the command if the new input low-order tributary belongs to a HO tributary which cannot be connected to the LO switch matrix as part of its substructuring.

Error code:

`IDNC – SARB – Status, All Resources Busy, LOXC capacity exhausted`

Validation rule: path-protected cross-connection has inputs from ports with same interface standard

If converting to a path-protected cross-connection then the according validation rule of ENT-CRS is executed.

Error code:

`SNVS – Status, Not in Valid State, Different interface standards for input tributaries`

Validation rule: No trib of any MSP port as *newinaid* for LO pathprotected cross-connection

If converting to a lower order pathprotected cross-connection, the command processing shall verify that the old tributary identified by *in_aid* and the new tributary identified by as *newinaid* do not belong to a 1+1 or 1:1 multiplex section/line protection group.

Error code:

`SNVS – Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO cross-connection`

For simplicity also the full validation rule of ENT-CRS may be executed.

Validation rule: No trib of any MSP 1x1 protection port as *newinaid* for HO pathprotected cross-connection

If converting to a higher order path protected cross-connection, the cross-connection command processing shall verify that the old tributary identified by *in_aid* and the new tributary identified by as *newinaid* do not belong to a protection port of a 1:1 multiplex section/line protection group.

Error code:

`SNVS – Status, Not in Valid State, trib of 1x1 protection port as input-trib in pathprotected cross-connection`

Validation rule: no multiple added XCs for MS-SPRing TOP

If converting to a path-protected cross-connection then the according validation rule of ENT-CRS is executed.

Error code:

`SNVS – Status, Not in Valid State, multiple added XCs for MS-SPRing TOP`

Validation rule: Supported MS-SPRing/BLSR DNI/DRI

If converting to a path-protected cross-connection then the according validation rule of ENT-CRS is executed.

Error code:

`SNVS – Status, Not in Valid State, unsupported MS-SPRing/BLSR DNI/DRI`

Validation rule: path-protected cross-connection: parameters missing

Cross-connection command processing shall deny the conversion to a pathprotected cross-connection if one of the mandatory parameters *htmr*, *newinaid*, *rme*, *snctype* or *wtr* is missing.

Error code: `IPMS -Input, Parameter Missing, <parameter name> missing`

Validation rule: No cross-connection loopback existing

If executing the **`CNVT-CCT`** command in order to establish a path protection then the new leg identified by the tributary newinaid must not have an existing cross-connection loopback.

Error code:

`SNVS – Status, Not in Valid State, new Tributary has existing loopback`

NOTE: There is no command validation concerning facility loopback i.e. the cross-connection establishing is possible even with a facility loopback present.

Validation rule: Tributaries not used in a test access session

If executing the **`CNVT-CCT`** command in order to establish a path protection then the new leg identified by the tributary newinaid and the tributaries involved in the existing cross-connection must not be used in a test access session either as tributary under test or as test access tributary.

Error code:
`SNVS – Status, Not in Valid State, tributaries used in test access session`

Validation rule: Restrict number of cross-connection using same input tributary

If converting to a path-protected cross-connection then the according validation rule of **`ENT-CRS`** is executed.

Error code:
`SNVS – Status, Not in Valid State, Too many crossconnection originating from same tributary`

Validation rule: BLSR/MS-SPRing protection access (Extra Traffic)

If converting to a path-protected cross-connection then the according validation rule of **`ENT-CRS`** is executed.

Error code:
`SNVS – Status, Not in Valid State, Command refers to protection tributaries of a BLSR/MS-SPRing protection`

Validation rule: *newloca* not provisionable for Extra Traffic

Cross-connection command processing shall deny commands, if cross-connections have protection tributaries of ports in BLSR/MSSPRing protection groups involved (Extra Traffic), and *newloca* is specified in the command.

Error code:
`IDNC – Input, Data Not Consistent, It is not allowed to specify loca/locz for EXTRA TRAFFIC`

Validation rule: No connections between BLSR/MS-SPRing working and protection tribs

If converting to a path-protected cross-connection then the according validation rule of **`ENT-CRS`** is executed.

Error code:
`IDNC – Input, Data Not Consistent, Connections between BLSR/MS-SPRing working and protection tributaries are not allowed`

Validation rule: No low-order connection with 4 fiber BLSR/MS-SPRing ports

If converting to a path-protected cross-connection then the according validation rule of **`ENT-CRS`** is executed.

Error code:
`IDNC – Input, Data Not Consistent, No low-order path protection with input tributaries from 4-fiber BLSR/MS-SPRing ports`

Validation rule: No low-order path protection input tributaries from the protection bandwith of 2-fiber MS-SPRing/BLSR protection

If converting to a path-protected cross-connection then the according validation rule of **ENT-CRS** is executed.

Error code:

`IDNC – Input, Data Not Consistent, No low-order path protection with input tributaries from 2 fiber MS-SPRing/BLSR protection`

Validation rule: SNC type compliance

If converting to a path-protected cross-connection then the according validation rule of **ENT-CRS** is executed.

Error code:

`IDNC – Input, Data, Not Consistent, SNC Type not compliant with existing pathprotected crossconnection`

Validation rule: No VCG-trib as *in_aid* (for pathprotected cross-connection)

If converting to a path-protected cross-connection then the according validation rule of **ENT-CRS** is executed.

Error code:

`IDNC – Input, Data, Not Consistent, VCG-trib as input-trib in pathprotected crossconnection`

Validation rule: Maximum number of *ppgnames* exceeded

If changing or entering the parameter *ppgname* then the according validation rule of **ENT-CRS** is executed.

Error code:

`SSRE – Status, System Resources Exceeded, limit of ppgnames exceeded`

Validation rule: No lower order TransMUX cross-connections when LO switch capacity is lower than 40G

When creating lower-order TransMUX cross-connections, the command processing shall verify that the low-order switching capacity of the LOXC pack in the system is greater or equal to 40 Gbit/s.

Error code:

`SNVS – SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Validation rule: Cross-connection must not be of *xconcls* NN

Cross-connection command processing shall verify, that the cross-connection is not of *xconcls* = NN.

Error code:

`SNVS – Status, Not in Valid State, Cross-connection must not be of xconcls NN`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* INNI

Cross-connection command processing shall verify, that a new input tributary does not belong to a port of *portClass* INNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass INNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of *portClass* INNI

For low-order cross-connections, cross-connection command processing shall verify that a new input tributary does not belong to a port of *portClass* INNI.

Error code:

`IDNC – Input, Data Not Consistent, Tributaries must not be on ports of portClass INNI`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* UNI

Cross-connection command processing shall verify, that a new input tributary does not belong to a port of *portClass* UNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass UNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of *portClass* UNI

For low-order cross-connections, cross-connection command processing shall verify that a new input tributary does not belong to a port of *portClass* UNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass UNI`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* ENNI

Cross-connection command processing shall verify, that a new input tributary does not belong to a port of *portClass* ENNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of *portClass* ENNI

For low-order cross-connections, cross-connection command processing shall verify that a new input tributary does not belong to a port of *portClass* ENNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI`

Validation rule (ONNS specific): Restrict number of crossconnection using same input tributary on EDGE port

Cross-connection command processing shall verify that the number of cross-connections originating from the same tributary is restricted to two, if the affected port is of port class EDGE.

Error code:

`SNVS – Status, Not in Valid State, Too many crossconnection originating from same tributary`

Related TL1 Messages

- "DLT-CRS: Delete cross-connection" (p. 3-145)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)
- "ED-CRS: Edit cross-connection" (p. 3-292)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).

CONN-TACC: Connect test access

General

The **CONN-TACC** command is used to initiate a test access session and to connect the test access tributaries.

Remote test access is *not* supported for

- SDH lower-order tributaries (VC-12, and lower-order VC-3).
- 1+1 MSP/APS, SNCP/UPSR and MS-SPRing/BLSR protection schemes

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **CONN-TACC**-*modifier:tid:tatrib1:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the connect test access command is to act.
   Type: Alphanumeric value set.
   Range:
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **tatrib1** – primary Test Access tributary
   AID of the primary Test Access tributary. The keyword **all** is not allowed on any level.
   Type:AID
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
4. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`etrib`** – AID of the E-tributary
   AID of the E-tributary for the test session.
   Type: AID.
   Range: GenTribAID or GenLOTribAID
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.
2. **`ftrib`** – AID of the F-tributary
   AID of the F-tributary for the test session. If there is no cross-connection at the E-tributary, this parameter is omitted.
   Type: AID.
   Range: GenTribAID or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.
3. **`tatrib2`** – secondary Test Access Tributary
   AID of the secondary test access tributary. In the case of test access modes with single test tributary the parameter is omitted.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.
4. **`tstmde`** – Test Access Mode
   Type: Alphanumeric value set.
   Range: Allowed values:
   - MONE – Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.
   - MONF – Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.
   - MONEF – Monitor E-end and F-end mode of test access. This test access mode monitors (E end and F end) both directions of a two-way cross-connection.

Output format

The NE returns the following normal completion response to a successful **`CONN-TACC`** command if there are no failure conditions to report:

```
   sid date time
M  ctag COMPLD
;
```

Detailed behavior description

The **`CONN-TACC`** command is used to establish a test access(es) connection to a tributary or tributaries in the NE.

If the test access session is established as MONE or MONF, the primary test tributary is used. If the test access session is established as MONEF, the primary test tributary is designated for accessing E-end incoming signal and the secondary test tributary is designated for accessing F-end incoming signal.

If there is no connection on E-end tributary, the F-end tributary AID parameter is omitted.

If there are one or more cross-connections established on E tributary, the F-end tributary AID parameter is required to be entered.

In addition to the **`CONN-TACC`** notification, an **`ED-CRS`** notification is sent to report the changed parameter values *taccstat* and *rtntaccstat*.

Update of affected entities when establishing a LO-test access session

When establishing a LO-test access session it may happen that the involved tributaries (LO test access tributaries or E-tributary or F-tributary) are not yet existing due to either a non-fitting substructure of the related HO-tributary or a missing HO-PTF.

In this case the **`CONN-TACC`** command shall create the necessary substructure or adapt an already existing substructure. For these side effects the same rules as for creating a LO-cross-connection is applied.

Note: Wrong error code when setting up a LO Remote Test Access session where not all involved tributaries are different.

Setting up a LO Remote Test Access (RTA) session is rejected if some or all of the AIDs of the primary and secondary test access tributary and the E-trib and F-trib are identical. The NE rejects such requests but uses a wrong error code. Instead of

`IDNV – Input, Data Not Valid, all tributaries must be different`

it returns

`IEAE – Input, Entity Already Exists, cross-connection exists on test tributary`

Example input/output

```
CONN-TACC-STS1:LT-UNITE:1-1-#-#-4-1-1:123456:::ETRIB=1-1-#-#-15-1-1,FTRIB=1-1-#-#-16
      -1-1,TSTMDE=MONE;
   LT-UNITE 02-10-01 16:39:52
M  123456 COMPLD
;
```

LO test session invocation:

```
CONN-TACC-LOVT1:LT-UNITE:1-1-#-#-4-1-1-28:123456:::\
ETRIB=1-1-#-#-15-1-1-28,FTRIB=1-1-#-#-16-1-1-28,TSTMDE=MONE;
   LT-UNITE 02-10-01 16:39:52
M  123456 COMPLD
```

;

Error responses

Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417)command error responses section. The error responses listed there also apply to the **CONN-TACC** command.

Validation rule: Missing F tributary

If the NE receives a **CONN-TACC** command in which there are one or more cross-connection using the E tributary as input and *ftrib* is not specified, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, missing F tributary`

Validation rule: E tributary and F tributary not cross-connected

If the NE receives a **CONN-TACC** command with an F tributary specified and there is no cross-connection between the E and the F tributaries, the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, E and F tributaries not cross-connected`

Validation rule: MONF or MONEF test access mode not allowed on a one-way cross-connection from E to F tributary

If the NE receives a **CONN-TACC** command with MONF or MONEF as test access mode when there is a one-way cross-connection on E to F tributaries, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid test access mode`

Validation rule: Cross-connection exists on the test tributary

If the NE receives a **CONN-TACC** command with a test tributary that has an existing cross-connection, the following error response is returned:

Error code:
`IEAE – Input, Entity Already Exists, cross-connection exists on test tributary`

Validation rule: Test tributary already connected

If the NE receives a **CONN-TACC** command with a test tributary that has an existing test access connection (to E and/or F tributary(s)) or if the **CONN-TACC** command is executed again with same or different attributes on the same test tributary, the following error response is returned:

Error code:
`IEAE – Input, Entity Already Exists, test access connection exists`

Validation rule: No test access session

Test Access command processing shall deny commands requesting to setup a test access session with a test tributary or E tributary or F tributary which is already used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: Test tributary or E or F tributary in a loopback

If the NE receives a **`CONN-TACC`** command with a test tributary or E or F tributary currently in a loopback, the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, Tributary has existing loopback`

Validation rule: Secondary test tributary not needed for MONE test modes

If the NE receives a **`CONN-TACC`** command with a secondary test tributary value for a MONE test access mode, the following error response is returned:

Error code:

```
INUP – Input, Non-null Unimplemented Parameter, secondary test
tributary not allowed
```

Validation rule: Pack restrictions for primary and secondary HO test access tributary

Test access command processing shall verify that the specified primary and secondary HO test access tributaries do not reside on:

- the OPT2G5 pack
- the OP40 pack
- the GE1 pack
- the GE10PL1 pack
- the TMUX2G5 pack
- the ES64U pack

Error code:

```
IDNC – Input, Data Not Consistent, primary/secondary test access
tributary not supported by that pack
```

Validation rule: Pack restrictions for primary and secondary LO test access tributary

Test access command processing shall verify that the specified primary and secondary LO test access tributaries do not reside on:

- the OPT2G5 pack
- the OP40 pack

- the EP51 pack (DS3 and EC1 interfaces)
- the GE1 pack
- the GE10PL1 pack
- an asynchronous ACG port of the TMUX2G5 pack
- the ES64U pack

Error code:

`IDNC – Input, Data Not Consistent, primary/secondary test access tributary not supported by that pack`

Validation rule: Pack restrictions for LO tributaries under test

The test access command processing verify that the specified primary and secondary LO test access tributaries do not reside on:

- the OPT2G5 pack
- the EP51 pack (DS3 and EC1 interfaces)
- the GE1 pack
- the GE10PL1 pack
- an asynchronous ACG port of the TMUX2G5 pack
- the ES64U pack

Error code:

`IDNC – Input, Data Not Consistent, LO tributary not supported by that pack`

Validation rule: Tributaries must exist for HO TACC sessions

If the network element receives a **CONN-TACC** command with a high-order test tributary E or F that does not exist, the following error response is returned:

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: No HO TACC sessions while LO cross-connection, substructure or conversion cross-connection exists

If the high-order tributaries specified in the high-order test access command are substructured or LO cross-connections or conversion cross-connections exist at the tributaries, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Entities must fit LOXC interface standard

Test access command processing shall verify that the specified modifier and the given aid's fit to the LOXC overall interface standard, i.e. the value of the parameter lostd related to the active LOXC pack.

Error code:

`IDNV – Input, Data Not Valid, Low-order test access sessions of this interface standard not supported`

Validation rule: Restrictions for LO SONET test access rates

For low-order SONET test access sessions command processing shall verify that the rates of the HO NIM/PTFs fit to the modifier parameter as follows: LOVT1 requires STS1 HO NIMs or TMSSTS1 HO PTFs.

Error code:

`IDNC – Input, Data Not Consistent, Specified Tributary does not support the indicated rate`

The existence of an STS1 HO NIM can be verified by checking the *inputsig*, *unequiposig* and *istd* parameters of the related port. Example: If the first HO NIM of a port needs to be successfully checked then the related port parameters should read as follows:

- *inputsig* starting with N1-… (N is a number of STS1 signals),
- *unequiposig* starting with N1-… (N is a number of STS1 signals),
- *istd* equals SONET.

The TMASTS1 HO PTFs exists on the TransMUX, if the parameter *tmuxcap* > 0.

Validation rule: HO tributaries must exist for LO TACC sessions

Cross-connection command processing shall verify that the high-order tributaries specified in the low-order CONN-TACC command do exist.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

The validation includes the check for provisioned circuit pack, optical module and port.

Validation rule: LO TACC sessions not at HO tributaries used in HO cross-connection

If the network element receives a **`CONN-TACC`** command with a low-order test tributary or E tributary or F tributary relating to a HO tributary which is involved in a HO cross-connection, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, HO cross-connection exists`

Validation rule: AID boundary and rate consistency

If the NE receives a **`CONN-TACC`** command with a test tributary or E or F tributary that is not consistent with the test access rate, the following error response is returned:

Error code:`IDNC – Input, Data, Not Consistent, Tributary boundary is inconsistent with the indicated rate`

Consistency implies the following condition:

- The boundary of the specified tributary is consistent with the specified rate
- STS-1/VC-3 payloads can start at any position
- The first (STS-1 equivalent) tributary of an STS-3/VC-4 rate cross-connection must be on a mod3 + 1 boundary (1, 4 ,7, …)
- The first STS-1 of an STS-12/VC-4-4c rate cross-connection must be on a mod12 + 1 boundary (1, 13, 25, …)
- The first STS-1 of an STS-48/VC-4-16c rate cross-connection must be on a mod48 + 1 boundary (1, 49, 97, …)
- The first STS-1 of an STS-192/VC-4-64c rate cross-connection must be on a mod192 + 1 boundary (1, 193, 385, …).

Validation rule: Tributary provisioning and rate consistency

Test access command processing shall verify that:

- if the port identified by logical input AID is provisioned to operate in the fixed signal rate mode, as determined by the tributary mode parameter, then the specified cross-connection rate is equivalent to the provisioned signal rate of the input tributary, as determined by the tributary input signal rate parameter.
- if the port identified by logical output AID is provisioned to operate in the fixed signal rate mode, as determined by the tributary mode parameter, then the specified cross-connection rate is equivalent to the provisioned signal rate of the output tributary, as determined by the tributary unequipped output signal rate parameter.

Note that in both cases it does not matter whether the connected tributaries are SONET (STS1/STS3/…) or SDH (VC3/VC4/…) signals with the same transmission rate.

Error code:

`IDNC – Input, Data Not Consistent, Specified Tributary does not support the indicated rate`

Validation rule: XC pack is (pre-)provisioned

If the NE receives a **`CONN-TACC`** command while no XC pack is pre-provisioned or provisioned, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, XC Pack not provisioned`

Validation rule: LOXC pack with more than 0G switch capacity

If the network element receives a **CONN-TACC** command for a low-order TACC session while no LOXC pack is provisioned with more than 0G LO switch capacity, then the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Validation rule: Not allowed for path protected tributaries

If the network element receives a **CONN-TACC** command with E- or F tributary AID belonging to a path protection, the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, Test Access not allowed with path protected tributaries`

Validation rule: Port used for MS-SPRing/BLSR

If the NE receives a **CONN-TACC** command with a test access tributary on a port which is used in an MS-SPRing/BLSR protection group, the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, port used for MS-SPRing/BLSR`

Validation rule: Port used for Linear MSP

If the NE receives a **CONN-TACC** command with a test access tributary on a port which is used in a Linear MSP protection group, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, port used for Linear MSP`

Validation rule: Not allowed for bridged cross-connections

If the network element receives a **CONN-TACC** command with E- or F-tributary used in a bridged cross-connection, the following error response is returned:

Error code:

`IEAE – Input, Entity Already Exists, cross-connection exists on test tributary`

Validation rule: Inconsistent tributary rate

If the NE receives a **CONN-TACC** command which does not match the connection rate of the connection between E and F, the following error response is returned:

Error code:

`IPNC – Input, Parameter Not Consistent, Rate inconsistent with crossconnection rate`

Validation rule: Port used for ONNS application

If the NE receives a **CONN-TACC** command with a test access tributary or a E or F tributary on a port with portClass other than TRADITIONAL the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, port used for NN application`

Validation rule: All tributaries must be different

If the NE receives a **CONN-TACC** command where two or more of the specified tributaries (test access tributaries, E or F tributary) are identical, the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, all tributaries must be different`

Validation rule: Spare capacity for HO cross-connection legs on LOXC

If the network element receives a **CONN-TACC** command for a low-order TACC session where at least one of the specified tributaries (test access tributaries, E or F tributary) requires a HO tributary to be a newly substructured and this HO tributary cannot be connected to the LO switch matrix because the limit of connectable VC4/STS1 connections is already reached, then the following error response is returned:

Error code: `SARB – Status, All Resources Busy, LOXC capacity exhausted`

Validation rule: No low-order TACC sessions with LOXC/1

If the network element receives a **CONN-TACC** command for a low-order TACC session, it shall verify that the low-order switching pack currently used by the system is not of type LOXC/1.

Error code:

`SNVS – Status, Not in Valid State, low-order test-access sessions not allowed with LOXC/1`

Related TL1 messages

- "RTRV-TACC: Retrieve test access" (p. 3-2974)
- "CHG-ACCMD: Change access mode" (p. 3-65)
- "DISC-TACC: Disconnect test access" (p. 3-129)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)

CONN-TACC notification

Output format

The **CONN-TACC** notification has the following specific syntax:
CONN-TACC-*modifier:tatrib1::spec_block;*

Output parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the connect test access command is to act.
   Type: Alphanumeric value set.
   Range:
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tatrib1** – primary Test Access Tributary
   AID of the primary Test Access tributary. The keyword **all** is not allowed on any level.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.

Spec block parameters

1. **etrib** – AID of the E-tributary
   AID of the E-tributary for the test session.
   Type: AID.
   Range: GenTribAID or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
2. **ftrib** – AID of the F-tributary
   AID of the F-tributary for the test session. If there is no cross-connection at the E-tributary, this parameter is omitted.
   Type: AID.
   Range: GenTribAID or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
3. **tatrib2** – secondary Test Access Tributary
   AID of the secondary Test Access Tributary. In the case of test access modes with single test tributary the parameter is omitted.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
4. **tstmde** – Test Access Mode
   Type: Alphanumeric value set.

Range:

- MONE (default): Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.
- MONF: Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.
- MONEF: Monitor E-end and F-end mode of test access. This test access mode monitors (E-end and F end) both directions of a two-way cross-connection.

Example input/output

```
CONN-TACC-STS1:LT-UNITE:1-1-#-#-4-1-1:123456:::ETRIB=1-1-#-#-15-1-1,FTRIB=1-1-#-#-16
      -1-1,TSTMDE=MONE;

   LT-UNITE 02-10-01 16:39:52
M  123456 COMPLD
;

   LT-UNITE 02-10-01 16:40:22
A  001 REPT DBCHG
   ″TIME=16-40-04,DATE=02-10-01:CONN-TACC-STS1:1-1-#-#-4-1-1::ETRIB=1-1-#-#-15-1-
      1,FTRIB=1-1-#-#-16-1-1,TSTMDE=MONE:″
;
```

On creation of lower-order test access sessions:

```
CONN-TACC-LOVT1:LT-UNITE:1-1-#-#-4-1-1-28:123456:::\
ETRIB=1-1-#-#-15-1-1-28,FTRIB=1-1-#-#-16-1-1-28,TSTMDE=MONE;

   LT-UNITE 02-10-01 16:39:52
M  123456 COMPLD
;

   LT-UNITE 02-10-01 16:40:22
A  001 REPT DBCHG
   ″TIME=16-40-04,DATE=02-10-01:CONN-TACC-LOVT1:1-1-#-#-4-1-1-28::\
    ETRIB=1-1-#-#-15-1-1-28,FTRIB=1-1-#-#-16-1-1-28,TSTMDE=MONE:″
;
```

COPY-RFILE: Copy remote file

The **COPY-RFILE** command can be used to initiate remotely a file transfer to/from an NE. The file transfer is intended to download a new software generics to the NE.

The actual file transfer can use the following file transfer applications:

- FTAM
- FTP
- FT-TD (file transfer translation device)
  and is determined by the use of the *src* and *fttd* parameters.

The **COPY-RFILE** was introduced on customer request. The differences to **CPY-MEM** are:

- A confirmation or rejection message is immediately returned by the system
- The progress and the result of the remote file transfer are reported by the autonomous message **REPT EVT FXFR**

The **COPY-RFILE** command can be used to download a database (DB DOWNLOAD). It can be activated in a separate step with the **APPLY** command.

Note: The support of a certain file transfer application is release specific and additionally dependent on the network configuration (e.g. when one GNE is configured to be a FT-TD, FT-TD can be used for file transfers to/from remote NEs).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 2

Abortable: No.

Input format

Command syntax: **CPY-RFILE***:tid::ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **type** – type and direction of the file transfer

This parameter pecifies the type and direction of the file transfer. The following file transfer types are supported:

- SWDL: SW download
- DBDL: Database download
- RFR: Remote file restore, restore the database from a backup location
- RFBU: Remote file backup, copy the database to a backup location

Type: Alphanumeric value set.
Range: SWDL, DBDL, RFR, RFBU.

2. **`src`** – the source of the file
   src is the source where the database or software is being transferred from.
   Type: text string;
   Range: 1 to 128 characters, format is a FTP URL or FTAM URL address on the system (EMS or CIT) from which the file is transferred
   The FTP URL must be in the format:
   ftp://[<login>[:<password>]@]<host>[:<port>]/<path_name>
   - <login>: default value: ″anonymous″
   - <password>: default for <login> ″anonymous″ is ″″ (empty). For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
   - <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0…255.
   - <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
   - <path_name>: The complete path name, where the files are located.

   The FTAM URL must be in the format: ftam://[<nsap_address>/<path_name>
   - <nsap_address>: network service access point address of the host, which consists of 20 bytes (40 hex characters)
   - <path_name>: The complete path name, where the files are located.

   Examples of valid path name are: ″/bwm/generic/01_02_15″, ″C:\\Program Files\\Alcatel-Lucent\\WaveStar″
3. **`dest`** – destination of the file
   dest is the destination where the database is being transferred to. The parameter is omitted if *type* = SWDL, DBDL, or RFR.
   src is the source where the database or software is being transferred from.
4. **`ovwrt`**
   This parameter indicates whether or not the files should overwrite an existing file located at the destination.
   Type: Alphanumeric value set.
   Range: YES (default)- Overwrite existing file of same directory location and filename. NO – Do not overwrite file, fail the file transfer if file already exists.

5. **`fttd`** – file transfer translation device
   This parameter specifies the file transfer translation device (FTTD).
   Type: text string.
   Range: 1 to 128 characters, format is a FTTD gateway address.
   The FTTD address must be in the format:
   - fttd://<nsap_address>
     <nsap_address>: network service access point address of the host, which consists of 20 bytes (40 hex characters). There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.

Output format

The NE sends the following normal completion response just prior to initiating the requested file transfer:

```
   sid date time
M  ctag COMPLD
;
```

Completion message if ES64U databases missing in backup

If one or more ES64U database backups could not be copied from the ES64U packs to the NVM before the complete database generic of the system are transferred to the management system, then the following completion response is given:

```
   sid date time
M  ctag COMPLD
   /* ES64U databases for following slots are missing: <List>*/
;
```

The <List> shall be a simple list of slot-numbers separated by commas.

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response. The output parameters listed there also apply to the **`COPY-RFILE`** command.

Detailed behavior description

Generation of REPT DBCHG message

No **`REPT DBCHG`** message is generated. Instead, **`REPT EVT FXFR`** notification is generated.

Program download capability

The **COPY-RFILE** command can be used to download software generics. The DOWNLOAD option of **COPY-RFILE** is used to download new generics which are stored in the previous region of NVM. A directory or URL is specified for the source (see *srcurl* parameter).

When the **COPY-RFILE** command is received by the NE, a confirmation or rejection must be returned immediately. The NE must ensure it recognizes the file/s it will be downloading and, when applicable, that there is enough space to store, uncompress, and install the file/s. So, the initial response will indicate the NEs imminent actions.

As an FTP client, the network element will use the information included in the TL1 command to establish an FTP session with Transport Element Activation Manager, being the FTP server.

Installation via **APPLY** causes the contents of previous NVM to be installed. Following installation, a ″pointer″ is changed so that what is identified as current NVM contains the currently executing generic and previous NVM contains the previous generic.

Database backup and restore capability

The **COPY-RFILE** command can be used to backup and restore the configuration database. The RFBU option of **COPY-RFILE** is used to backup the database, the RFR option is used to restore the database. A directory or URL is specified for the source (see *src* parameter) in the case of RFR or for the destination (see *dest* parameter) in the case of RFBU.

Backup with ES64U packs

All database of the ES64U cards shall first be copied from the ES64U packs to the NVM, then the complete database generic of the system shall be transferred to the management system . By this, the system shall first check if all ES64U databases could be retrieved. If this is not the case, the appropriate error message shall be added to the complete message.

Database download capability

The DBDL option of the **COPY-RFILE** command can be used to download a database from a source at the *WaveStar*® CIT or the management system. A directory or URL is specified for the source (see *src* parameter).

The database is copied from a file store into previous region of NVM.

After successful completion of the database download the system checks if ES64U databases are related to slots which are actually provisioned with ES64U packs. If this is the case, the state of the database in the previous NVM is DOWNLOADED. If this is not the case, the state is MISMATCH. If the DB Download does not successfully complete, then the state of the database in the previous NVM is CORRUPT.

The installation of the database from NVM to working memory has to be initiated after the download manually using the **APPLY** command.

File transfer requires a checksum

A file that is transferred via the **COPY-RFILE** command should contain one or more checksums to help validate that the file has not been corrupted.

Example input/output

The following examples show input and output of **COPY-RFILE** commands:

The following example shows a **COPY-RFILE** request that downloads a new generic from the manager to NVM of the NE by FTP with the help of a FTTP device. To start execution, the **APPLY** command must be used.

```
COPY-RFILE:LT-UNITE::123456:::TYPE=SWDL,
SRC=”ftp://user:pwd@128.218.128.281:21/vender/model/release1”,FTTD=”fttd://39840f80000000000000
OVWRT=YES;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a **COPY-RFILE** request that downloads a new generic from the manager to NVM of the NE by FTAM. To start execution, the **APPLY** command must be used.

```
COPY-RFILE:LT-UNITE::123456:::TYPE=SWDL,
SRC=”ftam://39840F80000000000000000000010a4fabc861d/C:\\Program Files\\Lucent
      Technologies\\WaveStar CIT\\generics\\susi030042”,OVWRT=YES;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a **COPY-RFILE** command that performs a database backup from the NE by FTP via FTTD to a Windows based FTP server.

```
COPY-RFILE:UNITE-NE::123456:::TYPE=RFBU,
DEST=”ftp://user:pwd@128.218.128.281:21/backups\\UNITE-
      NE\\080102”,FTTD=”fttd://39840f800000000000000000000601D0D50E01d”,
OVWRT=YES;
   UNITE-NE 08-01-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **COPY-RFILE** request that downloads a new generic from the manager to NVM of the NE by FTAM. To start execution, the **APPLY** command must be used.

```
COPY-RFILE:LT-UNITE::123456:::TYPE=SWDL,
SRC=”ftam://39840F80000000000000000000010a4fabc861d/C:\\Program Files\\Lucent
      Technologies\\WaveStar CIT\\generics\\susi030042”,OVWRT=YES;
```

```
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a **COPY-RFILE** request that downloads a database from the manager to NVM of the NE by FTP with the help of a FTTP device. To cause the installation as working database, the **APPLY** command must be used.

```
COPY-RFILE:LT-UNITE::123456:::TYPE=DBDL,
SRC=”ftp://user:pwd@128.218.128.281:21/configuration/backups/051215”,FTTD=”fttd://39840f80000
OVWRT=YES;
   LT-UNITE 05-12-16 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command error response section. The error responses listed there also apply to the **COPY-RFILE** command. Additional error responses that specifically apply to this command are defined in the following:

Validation rule: Input parameters not specified

If any one or more of the input parameter values are not valid, the system responds with:

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Insufficient file space

If the command failed due to insufficient file space, the system responds with:

Error code: `ENEQ – Equipage, Not Equipped, insufficient file space`

Validation rule: Temporary exhaustion of resources

If the system is experiencing temporary exhaustion of allocated resources because

- all resources are busy or
- a **CPY-MEM** request is issued and a previous **CPY-MEM** has not finished or a resource is busy for other reasons,

the system responds with:

Error code: `SARB – Status, All Resources Busy`

Validation rule: Directory/URL not specified

If **COPY-RFILE** specifies an SWDL and no directory/URL is specified by *src*, then the following error response is given:

Error code:

`ICNV – Input, Command Not Valid, directory/URL must be specified`

Validation rule: Directory/URL not specified for restoration

If **`COPY-RFILE`** specifies an RFR and no directory/URL is specified by *src*, then the following error response is given:

Error code:
`ICNV – Input, Command Not Valid, directory/URL must be specified`

Validation rule: Directory/URL not specified for backup

If **`COPY-RFILE`** specifies an RBU and no directory/URL is specified by *dest*, then the following error response is given:

Error code:
`ICNV – Input, Command Not Valid, directory/URL must be specified`

Validation rule: Directory/URL not specified for database download

If **`COPY-RFILE`** specifies a DBDL and no directory/URL is specified by *src*, then the following error response is given:

Error code:
`ICNV – Input, Command Not Valid, directory/URL must be specified`

Validation rule: FT-TD device not specified

If **`COPY-RFILE`** used FTP for SWDL and no FT-TD device (nsap) is specified by *fttd*, then the following error response is given:

Error code: `ICNV – Input, Command Not Valid, FTTD must be specified`

Validation rule: Additional directory/URL specified

If **`COPY-RFILE`** specifies an SWDL and *dest* is not empty:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: Additional directory/URL specified for restoration

If **`COPY-RFILE`** specifies an RFR and *dest* is not empty:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: Additional directory/URL specified for backup

If **`COPY-RFILE`** specifies an SWDL and *src* is not empty:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: Additional directory/URL specified for database download

If **`COPY-RFILE`** specifies a DBDL and *dest* is not empty:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: No SW download, DB download, or DB restore during trial phase

If **`COPY-RFILE`** specifies an SWDL, RFR, or DBDL and a newly applied SW generic is still in trial phase (previous SW generic state and database state = PRESERVED), the command is denied:

Error code:
`SNVS – Status, Not in Valid State, SW generic is in trial phase`

Validation rule: No DB restore with incompatible database

If **`COPY-RFILE`** specifies an RFR or DBDL, the system verifies that the database to be restored is compatible with the currently running SW generic and the actual HW equipage. The attempt to restore an incompatible database is denied with:

Error code: `SNVS – Status, Not in Valid State, database not compatible`

Validation rule: No DB restore with ES64U pack

If **`CPY-RFILE`** specifies a DB restore by RFR, the system verifies that ES64U pack is (pre-)provisioned in the system. If a ES64U pack is (pre-)provisioned in the system the NE refuses the command.

Error code:
`SNVS – Status, Not in Valid State, No restore (RFR) with ES64U pack`

Related TL1 messages

- “APPLY: Apply software generic or activate a downloaded database” (p. 3-35)
- “RTRV-PRMTR-DATA: Retrieve parameter data” (p. 3-2744)
- “RTRV-PRMTR-SFTWR: Retrieve parameter software” (p. 3-2753)
- “RTRV-NE: Retrieve NE” (p. 3-2549)
- “ED-NE: Edit NE” (p. 3-442)

CPY-MEM: Copy memory

The **`CPY-MEM`** command can be used to initiate a file transfer to/from an NE. The file transfer is intended to

- download a new software generics to the NE (SW download), or
- to backup the NE database (DB backup), or
- to restore the NE database (DB restore).

The actual file transfer can use one of the following file transfer applications:

- FTAM
- FTP
- FT-TD (file transfer translation device).

and is determined by the use of the *from* and *to* parameters.

The command **`CPY-MEM`** was the original command for file transfer to/from an NE. The differences to **`COPY-RFILE`** are:

- The confirmation or rejection message is returned by the system **after** the transfer.
- The interim progress of the remote file transfer are **NOT** reported by autonomous messages.

The **`CPY-MEM`** command can be used to download a database (DB download). The database can then be activated in a separate step with the **`APPLY`** command.

`CPY-MEM` utilizing the Transport Service Bridge (TSB) is not supported.

Note: The support of a certain file transfer application is release specific and additionally dependent on the network configuration (e.g. when one GNE is configured to be a FT-TD, FT-TD can be used for file transfers to/from remote NEs).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 2

Abortable: No

Input format

Command syntax: **`CPY-MEM`**:*tid*::*ctag*::*from*,[*src*],*to*,[*dst*],*class*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **`from`** – from memory type
   This parameter specifies the memory where the data is being transferred from.
   Type: text string.
   Range:
   - ″PRI″, PRImary NVM of the NE. Although there is no secondary NVM, PRI is used for compatibility reasons
   - ″CIT, *psel*, *ssel*, *tsel*, *sap*″, CIT means Craft Interface Terminal (CIT)
   - ″EMS, *psel*, *ssel*, *tsel*, *nsap*″, EMS means Element Management System (EMS)
   - ″FTTD, *psel*, *ssel*, *tsel*, *nsap*″ FTTD (file transfer translation device)
     sel, *ssel*, *tsel*, and *nsap* are part of the from (and to) parameter and specify NSAP and selectors of the CIT or EMS or the GNE in case of FTTD. The sizes of *sel*, *ssel*, *tsel*, and *nsap* are as follows: *psel* (path selector): 0-4 bytes (0-8 hex characters); *ssel* (session selector): 0-4 bytes (0-8 hex characters); *tsel* (transport selector): 0-16 bytes (0-32 hex characters); *nsap* (network service access point): 20 bytes (40 hex characters).

   There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.
4. **`src`** – source
   This parameter specifies the source where the database or software is being transferred from. The parameter is omitted or specified as empty string ("") if *class* = BACKUP.
   Type: text string
   Range: 1 to 128 characters, format is a valid complete directory name , or an FTP URL on the system (EMS or CIT) from which the file is transferred.
   The FTP URL must be in the format:
   ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>
   - <login>: default value: ″anonymous″
   - <password>: default for <login> ″anonymous″ is ″″ (empty).For security reasons, the <password> portion of the FTP URL format must be masked prior to generating any autonomous messages or entries in the Notification Log.
   - <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0…255.

- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

Because backslash (\) is the TL1 escape character, to include a backslash as part of a directory name, it must be entered as a double backslash. A double backslash counts as only one character. Craft personnel will not have to type the double backslash for the GUI. The second backslash is automatically generated.
Examples of valid file names are: ″d:\\bwm\\data\\file1.dat″ or ″/bwm/generic/01_02_15.prog″.

5. **`to`** – to memory type
   This parameter specifies the memory where the data is being transferred to.
   Type: text string.
   Range:
   - ″PRI″, PRImary NVM of the NE. Although there is no secondary NVM, PRI is used for compatibility reasons
   - ″CIT,*psel*,*ssel*,*tsel*,*sap*″, CIT means Craft Interface Terminal (CIT)
   - ″EMS,*psel*,*ssel*,*tsel*,*nsap*″, EMS means Element Management System (EMS)
   - ″FTTD, *psel*, *ssel*, *tsel*, *nsap*″, FTTD (file transfer translation device).

     sel, *ssel*, *tsel*, and *nsap* are part of the from (and to) parameter and specify NSAP and selectors of the CIT or EMS or the GNE in case of FTTD. The sizes of *sel*, *ssel*, *tsel*, and *nsap* are as follows: *psel* (path selector): 0-4 bytes (0-8 hex characters); *ssel* (session selector): 0-4 bytes (0-8 hex characters); *tsel* (transport selector): 0-16 bytes (0-32 hex characters); *nsap* (network service access point): 20 bytes (40 hex characters).

     There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.
6. **`dst`** – destination directory name
   This parameter reflects the destination where the database is being transferred to.
   The parameter is omitted or specified as empty string (“”) if *class* = DOWNLOAD, RESTORE, or DBDOWNLOAD.
   See parameter *src*. Those requirements also apply to this parameter *dst*.
7. **`class`** – Class Of The Copy
   Specifies the class of the copy.
   Supported classes:
   - DOWNLOAD: download an SW generic
   - BACKUP: copy the database to a backup location
   - RESTORE: restore the database from a backup location
   - DBDOWNLOAD: download a database from a backup location

   Type: Alphanumeric value set.

Range:

- DOWNLOAD: download an SW generic
- BACKUP: copy the database to a backup location
- RESTORE: restore the database from a backup location
- DBDOWNLOAD: download a database from a backup location

Output format

If the NE fully complies with the **CPY-MEM** command, then the NE sends the following normal completion response after the copy has been completed:

```
   sid date time
M  ctag COMPLD
;
```

Completion message if ES64U databases missing or superfluous

If ES64U databases are missing or superfluous in the database generic and the user has performed a DBDOWNLOAD or BACKUP command, then the following completion response is given:

```
   sid date time
M  ctag COMPLD
   /* ES64U databases for following slots are missing or superfluous:<List>*/
;
```

The <List> shall be a simple list of slot-numbers separated by commas. Note: This message is sent when DB state goes to MISMATCH.

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response. The output parameters listed there also apply to the **CPY-MEM** command.

Detailed behavior description

Generation of `REPT DBCHG` message

The **CPY-MEM**command generates a **REPT DBCHG** message.

Program download capability

The **CPY-MEM** command can be used to download software generics. The DOWNLOAD option of CPY-MEM is used to download new generics which are stored in the previous region of NVM.

A directory or URL is specified for the source (see *src* parameter).

Installation via APPLY causes the contents of previous NVM to be installed. Following installation, a ″pointer″ is changed so that what is identified as current NVM contains the currently executing generic and previous NVM contains the previous generic.

Backup capability

The BACKUP option of the **`CPY-MEM`** command can be used to backup the database from the current region of NVM to a destination which is located at the CIT or EMS.

A directory or URL is specified for the destination (see *dst* parameter).

If database of available ES64U packs cannot be backed upd for any reason, the “Completion Message, if ES64U Databases missing or superfluous” is used.

Restore capability

The RESTORE option of the **`CPY-MEM`** command can be used to restore the database from a source at the CIT or EMS.

A directory or URL must be specified for the source (see *src* parameter).

The database is copied from a file store into previous region of NVM and then installed as the working database.

There are two phases to database restoration: copying the database from the source (e.g. the CIT) to NVM and then installing the database from NVM to working memory. The first phase takes much longer than the second.

CPY-MEM with *class* RESTORE can only be executed in maintenance mode.

Restore causes database installation

If CPY-MEM is used to restore a database to NVM, the NE will return COMPLD before the CTL controller reset.

The automatic installation of the database for a RESTORE is in contrast with the manual installation (via APPLY) of software required after a DOWNLOAD.

After the database has been installed to working memory, the working database contains provisioning that was in the restoration directory.

Restore clears logs

The system shall clear all Logs except the Security Activity Log when a database is successfully restored.

File transfer requires a checksum

A file that is transferred via the **`CPY-MEM`** command should contain one or more checksums to help validate that the file has not been corrupted.

Restore uses previous NVM

When CPY-MEM is used to restore a database, the previous region of NVM is used as an intermediate storage.

This minimizes the effect of copying to NVM since the current region of NVM is still available for database updates. Also, if communications fail during the copy to previous NVM, the current database is still available.

Provisioned vs. actual configuration

After a database restoration, if there is a discrepancy between the actual and provisioned CPs, then an alarm will be issued telling the craft about the discrepancy, but the NVM will not be altered automatically.

Example of discrepancy: The hardware indicates that an OC-48 pack is in slot 6, while the provisioned values in NVM indicate that a OC-192 pack should be in that slot. A CP mismatch alarm will be raised.

Database download capability

The DBDOWNLOAD option of the **CPY-MEM** command can be used to download a database from a source at the CIT or EMS. A directory or URL must be specified for the source (see *src* parameter). The database is copied from a file store into previous region of NVM.

After successful completion of the database download, the system checks if ES64U databases are contained in the database generic or ES64U packs are available in the system.

Afterwards the system checks if the contained ES64U data bases fit to the current configuration. If this is the case, the state of the database in the previous NVM is DOWNLOADED and the “Normal completion message” is used. If this is **NOT** the case, the state is MISMATCH and “Completion Message, if ES64U Databases missing or superfluous” is used.

If the DB download does not successfully complete, then the state of the database in the previous NVM is CORRUPT,.

The installation of the database from NVM to working memory has to be initiated after the download manually using the **APPLY** command.

Example input/output

The following examples show input and output of **CPY-MEM** commands.

The following example shows a **CPY-MEM** request that downloads a new generic from the EMS to PRI. To cause *prgm* to execute, the **APPLY** command must be used.

```
CPY-MEM:LT-
      UNITE::123456::”EMS,02,04,02,39840F80000000000000000000008006A19784500”,”/unite/release1
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command that downloads a database from a directory of the CIT PC to PRI:

```
CPY-MEM:LT-
      UNITE::123456::”CIT,01,01,03,39840F80000000000000000000008006A13244500”,”\\data\\00_11_01”,
   LT-UNITE 00-11-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command that backups the database in NVM to the CIT and stores it in a directory named data in a location named 00_11_01.

```
CPY-MEM:LT-
      UNITE::123456::”PRI”,,”CIT,01,01,03,39840F80000000000000000000008006A13244500”,”\\data\\00_
   LT-UNITE 00-11-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command that restores the database from the CIT:

```
CPY-MEM:LT-
      UNITE::123456::”CIT,01,01,03,39840F80000000000000000000008006A13244500”,”\\data\\00_11_01”,
   LT-UNITE 00-11-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command used for performing a database backup via the FT-TD to an FTP server (UNIX). The *psel*, *ssel*, *tsel*, *nsap* values identify the GNE which runs the FT-TD function:

```
CPY-MEM:LT-UNITE::123456::
      ”PRI”,,”FTTD,01,01,03,39840F80000000000000000000008006A132445500”,”ftp://johndoe:passwd@ftp
   LT-UNITE 00-11-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command used for performing a software download via the FT-TD from an FTP server (UNIX). The *psel*, *ssel*, *tsel*, *nsap* values identify the GNE which runs the FT-TD function:

```
CPY-MEM:LT-UNITE::123456::
      ”FTTD,01,01,03,39840F80000000000000000000008006A132445500”,,”PRI”,”ftp://johndoe:passwd@ftp
   LT-UNITE 00-11-02 08:00:00
M  123456 COMPLD
;
```

The following example shows a **CPY-MEM** command used for performing a database download via the FT-TD from an FTP server (Windows). The *psel*, *ssel*, *tsel*, *nsap* values identify the GNE which runs the FT-TD function:

```
CPY-MEM:UNITE-NE::123456::
      ”FTTD,01,01,03,39840F80000000000000000000008006A132445500”,,”PRI”,”ftp://johndoe:passwd@f
   LT-UNITE 08-01-02 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command error response section. The error responses listed there also apply to the **CPY-MEM** command. Additional error responses that specifically apply to this command are defined in the following:

Validation rule: Input parameters not specified

If any one or more of the input parameter values are not valid, the system responds with:

Error code: `IDNV – Input, Data Not Valid`

The following parameter combinations are allowed based on the file transfer application:

	class	from	src	to	dst
FTAM	DOWNLOAD	”CIT,psel,ssel,tsel,nsap”; ”EMS,psel,ssel,tsel,nsap”	<dir>	”PRI”	-
	BACKUP	”PRI”	-	”CIT,psel,ssel,tsel,nsap”; ”EMS,psel,ssel,tsel,nsap”	<dir>
	RESTORE	”CIT,psel,ssel,tsel,nsap” ”EMS,psel,ssel,tsel,nsap”	<dir>	”PRI”	-
FT-TD	DOWNLOAD	”FTTD,psel,ssel,tsel,nsap”	<URL>	”PRI”	-
	BACKUP	”PRI”	-	”FTTD,psel,ssel,tsel,nsap”;	<URL>
	RESTORE	”FTTD,psel,ssel,tsel,nsap”	<URL>	”PRI”	-

The parameter combinations allowed for the class RESTORE are also valid for the option DBDOWNLOAD.

Validation rule: Invalid file or directory name

If the command failed due to

- an invalid directory or
- an invalid source file or
- file store not found or
- file access denied,

the system responds with:

Error code: `SDNC – Status, Data Not Consistent, invalid file or directory`

Validation rule: Source product type does not match network product type

If a product type of the source does not match the product type of the NE, the system responds with:

Error code: `SDNC – Status, Data Not Consistent`

Validation rule: Insufficient file space

If the command failed due to insufficient file space, the system responds with:

Error code: `ENEQ – Equipage, Not Equipped, insufficient file space`

Validation rule: Product mismatch

If the command failed due to a product mismatch in a retrieved file (file does not belong to NE type), the system responds with:

Error code:
`SRAC – Status, Requested Access Configuration invalid, retrieved file NE mismatch`

Validation rule: Temporary exhaustion of resources

If the system is experiencing temporary exhaustion of allocated resources because

- all resources are busy or
- a CPY-MEM request is issued and a previous CPY-MEM has not finished or a resource is busy for other reasons,

the system responds with:

Error code: `SARB – Status, All Resources Busy`

Validation rule: External communication failure

If the command failed due to an external communications failure, the system responds with:

Error code:
`SROF – Status, Requested Operation Failed, external communications failure`

The following list identifies conditions that will cause an SROF error response as given above:

- FTAM error
- External communication failure
- Unsuccessful file transfer

Validation rule: Backup failed after unsuccessful retransmission of files

If during a database backup a transaction log overflow occurs, and the retransmission of database files fails, then the command terminates with the error:

Error code:
`SROF – Status, Requested Operation Failed, unsuccessful file retransmission`

During database backup all database changes caused by provisioning activities are stored temporarily in a transaction log. This prevents changes of the database while the backup is running. Excessive provisioning during database backup may cause an overflow of the transaction log. In this case, the changes are written to the database and the affected database files are retransmitted. In the case that the retransmission fails, the backup is terminated with an error message.

Validation rule: Generic not available

If the command failed because the generic is not available, the system responds with:

Error code: `ENSG – Equipage, Not Software Generic, generic not available`

The following condition will cause this ENSG error response:

- Not a software generic file.

Validation rule: Directory not accessible

If a directory is specified that cannot be found or cannot be accessed due to improper privileges, the system responds with:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Deemed source file

If a source file is deemed invalid for miscellaneous reasons such as incorrect checksum, invalid control information, etc., the system responds with:

Error code: `SNVS – Status, Not in Valid State`

Validation rule: Directory/URL not specified

If **`CPY-MEM`** specifies a DOWNLOAD, RESTORE, or DBDOWNLOAD and no directory/URL is specified in *src* or if it specifies a BACKUP and no directory/URL is specifies in *dst*, then the following error response is given:

Error code: `ICNV – Input, Command Not Valid, file must be specified`

Validation rule: No SW download, DB download or DB restore during trial phase

If **`CPY-MEM`** specifies a DOWNLOAD, RESTORE, or DBDOWNLOAD and a newly applied SW generic is still in trial phase (previous SW generic state and database state = PRESERVED), the command is denied:

Error code:

`SNVS – Status, Not in Valid State, SW generic is in trial phase`

Validation rule: No DB restore with incompatible database

If **`CPY-MEM`** specifies a RESTORE or DBDOWNLOAD, the system verifies that the database to be restored is compatible with the currently running SW generic and the actual HW equipage. The attempt to restore an incompatible database is denied with:

Error code: `SNVS – Status, Not in Valid State, database not compatible`

Validation rule: No DB restore with ES64U Pack

If **`CPY-MEM`** specifies a RESTORE, the system verifies that no ES64U pack is (pre-)provisioned in the system. If an ES64U pack is (pre-)provisioned in the system, the NE refuses the command.

Error code: `SNVS – Status, Not in Valid State, No restore with ES64U pack`

Related TL1 messages

- "APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
- "RTRV-PRMTR-DATA: Retrieve parameter data" (p. 3-2744)
- "RTRV-PRMTR-SFTWR: Retrieve parameter software" (p. 3-2753)
- "RTRV-NE: Retrieve NE" (p. 3-2549)
- "ED-NE: Edit NE" (p. 3-442)

CPY-MEM notification

Output format

Command syntax: **CPY-MEM**::*from*,[*src*],*to*,[*dst*],*class*;

Output parameters

Command parameters

1. **from** – from memory type
   Specifies the memory where the data is being transferred from.
   Type: text string.
   Range:
   - ″PRI″, PRImary NVM of the NE. Although there is no secondary NVM, PRI is used for compatibility reasons
   - ″CIT, *psel*, *ssel*, *tsel*, *sap*″, CIT means Craft Interface Terminal (CIT)
   - ″EMS, *psel*, *ssel*, *tsel*, *nsap*″, EMS means Element Management System (EMS)
   - ″FTP″, FTP means FTP Server
   - ″FTTD, *psel*, *ssel*, *tsel*, *nsap*″ FTTD (file transfer translation device)
     sel, *ssel*, *tsel*, and *nsap* are part of the from (and to) parameter and specify NSAP and selectors of the CIT or EMS or the GNE in case of FTTD. The sizes of *sel*, *ssel*, *tsel*, and *nsap* are as follows: *psel* (path selector): 0-4 bytes (0-8 hex characters); *ssel* (session selector): 0-4 bytes (0-8 hex characters); *tsel* (transport selector): 0-16 bytes (0-32 hex characters); *nsap* (network service access point): 20 bytes (40 hex characters).

   There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.
2. **src** – source
   This parameter reflects the source where the database or software is being transferred from. The parameter is omitted or specified as empty string (“”) if *class* = BACKUP.
   Type: text string
   Range: 1 to 128 characters, format is a valid complete directory name, or an FTP URL on the system (EMS or CIT) from which the file is transferred.
   The FTP URL must be in the format:
   ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>
   - <login>: default value: ″anonymous″
   - <password>: default for <login> ″anonymous″ is ″″ (empty).For security reasons, the <password> portion of the FTP URL format must be masked prior to generating any autonomous messages or entries in the Notification Log.
   - <host>: the <host> portion is an internet address of the form a.b.c.d. with a, b, c and d in the range of 0…255.

- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

Because backslash (\) is the TL1 escape character, to include a backslash as part of a directory name, it must be entered as a double backslash. A double backslash counts as only one character. Craft personnel will not have to type the double backslash for the GUI. The second backslash is automatically generated.
Examples of valid file names are: ″d:\\bwm\\data\\file1.dat″ or ″/bwm/generic/01_02_15.prog″.

3. **`to`** – to memory type
Specifies the memory where the data is being transferred to.
Type: text string.
Range:
   - ″PRI″, PRImary NVM of the NE. Although there is no secondary NVM, PRI is used for compatibility reasons
   - ″CIT,*psel*,*ssel*,*tsel*,*sap*″, CIT means Craft Interface Terminal (CIT)
   - ″EMS,*psel*,*ssel*,*tsel*,*nsap*″, EMS means Element Management System (EMS)
   - FTTD, *psel*, *ssel*, *tsel*, *nsap*″, FTTD (file transfer translation device)
   - ″FTP″, FTP means FTP Server.

     sel, *ssel*, *tsel*, and *nsap* are part of the *from* (and to) parameter and specify NSAP and selectors of the CIT or EMS or the GNE in case of FTTD. The sizes of *sel*, *ssel*, *tsel*, and *nsap* are as follows:
     psel (path selector): 0-4 bytes (0-8 hex characters);
     ssel (session selector): 0-4 bytes (0-8 hex characters);
     tsel (transport selector): 0-16 bytes (0-32 hex characters);
     nsap (network service access point): 20 bytes (40 hex characters).
     There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.
4. **`dst`** – destination directory name
Reflects the destination where the database is being transferred to.
The parameter is omitted or specified as empty string (“”) if *class* = DOWNLOAD or RESTORE.
src is the source where the database or software is being transferred from.
Type: text string
Range: 1 to 128 characters, format is a valid complete directory name, or an FTP URL on the system (EMS or CIT) from which the file is transferred.
The FTP URL must be in the format:
ftp://[<login>[:<password>]@]<host>[:<port>]/<file name>
   - <login> – default value: ″anonymous″
   - <password> default for <login> ″anonymous″ is ″″ (empty).

     For security reasons, the <password> portion of the FTP URL format is masked prior to generating any autonomous messages or entries in the Notification Log.

- <host> – The <host> portion is an internet address of the form a.b.c.d with a,b,c and d in the range of 0…255.
- <port> - This is the port number of the FTP server, given when deviating from the standard port number 21.
- <file name> – The complete path name of the file. Because backslash (\) is the TL1 escape character, to include a backslash as part of a directory name, it must be entered as a double backslash. A double backslash counts as only one character. Craft personnel will not have to type the double backslash for the GUI. The second backslash is automatically generated.

  Examples of valid file names are : ″d:\\bwm\\data\\file1.dat″ or ″/bwm/generic/01_02_15.prog″

5. **`class`** – class of the copy
   Specifies the class of the copy.

   The supported classes are:

   - DOWNLOAD: download the program
   - BACKUP: copy the database to a backup location
   - RESTORE: restore the database from a backup location

   Type: Alphanumeric value set.
   Range: DOWNLOAD, BACKUP or RESTORE

Example output

The following is an example for a CPY-MEM notification after a software download:

```
   LT-UNITE 00-11-01 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:CPY-
      MEM::\″EMS,02,04,02,39840F80000000000000000000008006A19784500\″,\″/unite/release1/prgm\″
;
```

The following is an example for a CPY-MEM notification after a DB restore (before the NE resets):

```
   LT-UNITE 00-11-01 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:CPY-
      MEM::\″CIT,01,01,03,39840F80000000000000000000008006A13244500\″,\″\\data\\00_11_
      01\″,\″PRI\″,,RESTORE″
;
```

The following is an example for a CPY-MEM notification after a DB backup via the FT-TD:

```
   LT-UNITE 00-11-01 15:00:00
A  000 REPT DBCHG
```

```
  ”TIME=15-00-00,DATE=00-11-03:CPY-
     MEM::\”PRI\”,,\”FTTD,01,01,03,39840F80000000000000000000008006A132445500\”,\”ftp://johndoe:
;
```

DISC-TACC: Disconnect test access

General

The **DISC-TACC** command is used to remove the test access connection and terminate the test access session on the test access tributary in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **DISC-TACC-modifier***:tid:tatrib1:ctag;*

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the connect test access command is to act.
   Type: Alphanumeric value set.
   Range:
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **tatrib1** – primary Test Access tributary
   AID of the primary Test Access tributary. The keyword **all** is not allowed on any level.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

The NE returns the following normal completion response to a successful DISC-TACC command if there are no failure conditions to report:

```
   sid date time
M  ctag COMPLD
;
```

Detailed behavior description

The **DISC-TACC** command is used to terminate a test access session. All s to the test access tributaries will be removed and the cross-connections between the E and F tributaries are reestablished.

In addition to the DISC-TACC Notification an ED-CRS notification is sent to report the changed parameter values *taccstat* and *rtntaccstat*.

Update of affected entities when removing a LO-test access session

When removing a LO-test access session it may happen that the HO-tributaries related to the involved LO-tributaries (LO test access tributaries or E-tributary or F-tributary) have to be de-substructured again. In this case the **DISC-TACC** command shall remove the substructure following the same rules as when deleting the last lower-order cross-connection.

Example input/output

```
DISC-TACC-STS1:LT-UNITE:1-1-#-#-4-1-1:123456;
   LT-UNITE 02-10-01 17:15:08
M  123456 COMPLD
;
```

For lower-order test access sessions:

```
DISC-TACC-LOVT1:LT-UNITE:1-1-#-#-4-1-1-28:123456;
   LT-UNITE 02-10-01 17:15:08
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DISC-TACC** command.

Validation rule: No test access session on the test tributary

If the NE receives a DISC-TACC command when there is no test access session set up on the test tributary specified, the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, no test access session on the test tributary`

Validation rule: Inconsistent tributary rate

If the NE receives an DISC-TACC command which does not match the test access connection rate, the following error response is returned:

Error code:

```
IPNC – Input, Parameter Not Consistent, Rate inconsistent with
crossconnection rate
```

Related TL1 messages

- “CONN-TACC: Connect test access” (p. 3-94)
- “CHG-ACCMD: Change access mode” (p. 3-65)
- “RTRV-TACC: Retrieve test access” (p. 3-2974)
- “RTRV-CRS: Retrieve cross-connection” (p. 3-2306)

DISC-TACC notification

Output format

The **DISC-TACC** notification has the following specific syntax:
DISC-TACC-*modifier:tatrib1::spec_block;*

Output parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the connect test access command is to act.
   Type: Alphanumeric value set.
   Range:
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tatrib1** – primary Test Access tributary
   AID of the primary Test Access tributary. The keyword **all** is not allowed on any level.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.

Spec block parameters

1. **etrib** – AID of the E-tributary
   AID of the E-tributary for the test session.
   Type: AID.
   Range: GenTribAID or GenLOTribAID
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
2. **ftrib** – AID of the F-tributary
   AID of the F-tributary for the test session. If there is no cross-connection at the E-tributary, this parameter is omitted.
   Type: AID.
   Range: GenTribAID or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.
3. **tatrib2** – secondary Test Access Tributary
   AID of the secondary Test Access Tributary. In the case of test access modes with single test tributary the parameter is omitted.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **all** or **usall** or **vall**-values at any position of the aid is not allowed.

Example input/output

```
DISC-TACC-STS1:LT-UNITE:1-1-#-#-4-1-1:123456;

   LT-UNITE 02-10-01 17:15:08
M  123456 COMPLD
;

   LT-UNITE 02-10-01 17:15:38
A  001 REPT DBCHG
   "TIME=17-15-20,DATE=02-10-01:DISC-TACC-STS1:1-1-#-#-4-1-1:: ETRIB=1-1-#-#-15-1-
      1,FTRIB=1-1-#-#-16-1-1:"
;
```

For lower-order test access sessions:

```
DISC-TACC-LOVT1:LT-UNITE:1-1-#-#-4-1-1-28:123456;

   LT-UNITE 02-10-01 17:15:08
M  123456 COMPLD
;

   LT-UNITE 02-10-01 17:15:38
A  001 REPT DBCHG
   "TIME=17-15-20,DATE=02-10-01:DISC-TACC-LOVT1:1-1-#-#-4-1-1-28::\
    ETRIB=1-1-#-#-15-1-1-28,FTRIB=1-1-#-#-16-1-1-28:"
;
```

DLT-ASAP-PROF: Delete alarm severity assignment profile

The **DLT-ASAP-PROF** command is used to delete an ASAP. Default ASAP's cannot be deleted.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-ASAP-PROF**:*tid*::*ctag*::*pftype*,*pfname*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen ("-") is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN

- SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

4. **`pfname`** – ASAP Profile Name
   The name distinguishes ASAPs of the same ASAP type specified by *pftype*. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen ("-") is allowed in the name additionally.
   Range: length 1-24 characters.
   For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Output format

The following example shows the successful completion of the **`DLT-ASAP-PROF`** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`DLT-ASAP-PROF`** command deletes the ASAP specified by pftype and pfname. A profile must not be used when it will be deleted.

The modification is reported via the DLT-ASAP-PROF notification.

Example input/output

The following example shows the successful completion of the **`DLT-ASAP-PROF`** command by the NE if the pfname will be changed:

```
DLT-ASAP-PROF:LT-UNITE-123::123456::PORT,NUERNBERG;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-ASAP-PROF** command.

Validation rule: *pfname* does not exist

An ASAP with the name *pfname* does not exist.

Error code:
`SDNC – Status, Data Not Consistent, invalid instance of profile`

Validation rule: ASAP in use

The addressed ASAP is still in use.

Error code: `SDNC – Status, Data Not Consistent, profile in use`

Validation rule: *pfname* is default

The addressed ASAP is a default one which can not be deleted.

Error code: `IDNC – Input, Data, Not Consistent, default profile`

Related TL1 messages

- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “ENT-ASAP-PROF: Enter alarm severity assignment profile” (p. 3-1446)
- “RTRV-ASAP-PROF: Retrieve ASAP profile” (p. 3-2249)
- “RTRV-ASAP-ASGNMT: Retrieve ASAP assignment” (p. 3-2244)

DLT-ASAP-PROF notification

Output format

Syntax: The **`DLT-ASAP-PROF`** notification has the following specific syntax:

`DLT-ASAP-PROF`::*pftype*,*pfname*

Output parameters

Command parameters

1. **`pftype`** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.

   Range:

   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring

- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

2. **pfname** – ASAP Profile Name
The name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
Range: length 1-24 characters.
For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Example output

The following example shows a normal notification after a **DLT-ASAP-PROF** command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:DLT-ASAP-PROF::ENET,NUERNBERG″
;
```

DLT-CONN: Delete network connection

Overview

The **DLT-CONN** command is used for a graceful deletion of a network connection.

Graceful network connection deletion

The **DLT-CONN** command allows a graceful network connection deletion from the originator node. This command shall trigger the release of all resources of one or multiple network connections via all impacted domains. A forced deletion can be done via **DLT-PATH**.

Graceful network connection deletion by connID

A defined *connID* shall trigger a graceful deletion of the network connection associated with the defined identity.

Graceful network connection deletion by callID

A defined *callID* without defining *connID* shall trigger a graceful deletion of all network connections associated with the selected *callID*. A defined *callID* together with a defined *connID* shall trigger a graceful deletion of the network connection associated with the selected *connID*.

Graceful deletion only from originator node

The **DLT-CONN** command is only allowed on the originator node of the affected network connection.

A REPT CONNCHG notification is generated

The **DLT-CONN** command shall generate a **REPT CONNCHG** message for every removed Network Connection.

A REPT PTHCHG notification is generated

The **DLT-CONN** command shall generate a **REPT PTHCHG** message for every removed ONNS Connection.

Note

In case setup or deletion of many network connections over several domains are running and reboots of ingress or egress nodes in affected domains occur at the same time, some remaining network connection segments may be left over.

In case during graceful deletion of a network connection over several domains some of the following unexpected events occur, some pieces of the network connection may remain undeleted:

- Failures of ENNI signalling channels between affected domains
- Reboots of ingress or egress nodes in affected domains

Remaining network connection segments can be identified by applying "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294) or "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) commands to affected ingress nodes. Such pieces then can be removed by applying a forced deletion via "DLT-PATH: Delete ONNS path connection" (p. 3-222) in one or all domains affected.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: NA

Abortable: No

Input format

Command syntax: **`DLT-CONN`**:*srcTid:srcAid:ctag:[callID]:[connID][:specBlock];*

Input parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. **`srcAid`**
   Source node ONNS application access identifier *srcAid* of the ONNS controller OS command interface port.
   Type: AID.
   Range: snn
   Default: n/a.

3. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.

4. **`callID`**
   The *callID* indicates a call of a logical call / connection separation (call capability). This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 … F).
5. **`connID`** – Connection ID
   Connection ID for an existing ONNS connection segment of a UNI or Network connection. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.

Output format

Normal system response to DLT-CONN for a network connection

If the **`DLT-CONN`** request is accepted by the system, the normal completion response is returned as shown below:

```
   srcTid date time
M  ctag COMPLD
   ″CallID=called[,ConnID=connID]″
;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date

3. **`time`**
   Time. The time parameter is set to the time when the output message is generated. See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59
   Default: current time
4. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.
5. **`completion code`**
   Standard TL1 completion code result.
   Type: Value set.
   Range:
   - COMPLD (default): for success
   - DENY: in case of errors
6. **`callID`**
   The *callID* indicates a call of a logical call / connection separation (call capability). This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0…F).
7. **`connID`** – Connection ID
   Connection ID for an existing ONNS connection segment of a UNI or Network connection. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.
   The output parameter*connID* is optional. It is omitted in case of multiple connections deleted under one *callID* with a single **`DLT-CONN`** command.

Example input/output

Input/output example of deleting a Network Connection

The following example shows the response of an accepted DLT-CONN command:

```
DLT-CONN:LT_ONNS_789:snn:433456:A123B3EF134A123B3EF13455:135.7.9.2-46:
192.1.165.12&1-6-5-3-1&1;

   LT-ONNS-789 2002-07-01 08:00:00
M  433456 COMPLD
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=192.1.165.12&1-6-5-3-1&1″
;
```

Error responses

Validation rule: Input Parameter *connID* not valid

If the value for the input parameter *connID* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid connection ID`

Validation rule: Input Parameter *callID* not valid

If the value for the input parameter *callID* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid callID`

Validation rule: Input Parameters not matching

If the value for the input parameters *connID* and *callID* are not matching, the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, input parameters are not matching`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT-PTHCHG notification" (p. 3-2149)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)

- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT CONNCHG notification" (p. 3-2104)

DLT-CRS: Delete cross-connection

The **DLT-CRS.***modifier* command can be initiated by a user to delete a cross-connection in the NE.

The *modifier* in the **DLT-CRS** command indicates the cross-connection rate of the tributary for which the command is applied to.

There is no difference if the rate is specified as a SONET rate (STS1, STS3,…) or as SDH-rate (VC3, VC4,…) for the deletion of the cross-connection i.e. the DLT-CRS-<*sonet-rate*> yields to the same physical and logical behavior as DLT-CRS-<*sdh-rate*>.

No check is made if the modifier matches the interface-standard of the port i.e. if the port is provisioned as SONET (resp. SDH) and the rate is a SONET (resp. SDH) rate.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **DLT-CRS**-*modifier*:*tid*:*in_aid*,*out_aid*:*ctag*::[*cct*],[*in_aid2*];

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **DLT-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1

- STS1
- STS3
- STS12
- STS48
- STS192
- AU3TU3
- TU3AU3

No use of the *all* value is allowed.

2. **tid** – Target Identifier
   Target Identifier. Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
3. **in_aid** – Input Access Identifier
   This parameter specifies the source of the cross-connection.
   In case of a 1-way cross-connection the source of a cross-connection is unambiguous. In this case the cross-connection will be retrieved by the RTRV-CRS, which has the identical *in_aid*.
   In case of a 2-way cross-connection the source is not defined (the source could be the *in_aid* or the *out_aid* of the cross-connection). In this case both aids (*in_aid* or *out-aid*) of the cross-connection can be used as the *in_aid* parameter of the **RTRV-CRS** command. The **RTRV-CRS** command will then return the *in_aid* – parameter used in the **RTRV-CRS** command as the *in_aid* of the RTRV-CRS reply.
   Example:
   ENT-CRS-VC4:NEname:1-1-#-#-1-1-1,1-1-#-#-2-1-1:123456:2way;
   RTRV-CRS-VC4:NEname:1-1-#-#-2-1-1:123457;
   M 123457 COMPLD ″1-1-#-#-1-1-,1-1-#-#-2-1-1:vc4…;
   Type: GenTribAID.
   Range: The **all**-parameter is accepted for the slot-, port- and trib-part of the AID. For the port the all-parameter is to be distinguished between **all** when referring OP***n*** packs and GE***n*** **vall** when referring the GE***n***.
   For example the following *in_aids* are valid:
   - 1-1-#-#-all-all-all specifying all tributaries and all ports in the whole shelf
   - 1-1-#-#-2-1-all specifying all tributaries of the first port of the pack in slot-number=02
   - 1-1-#-#-3-vall-all specifying all tributaries of the first port of the (GE***n***) pack in slot-number=03
4. **out_aid** – Output Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   Type: STS1NumAID (when using OP***n*** packs) or VCGNumAID (when using GE***n***).
   Range: n.a. (**all** values shall never occur as part of an output parameter).
5. **ctag** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

6. **`cct`** – cross-connection Topology
   This parameter specifies the topology of the cross-connection to be established via the **`ENT-CRS`** command.
   The topology defines
   - the number of legs of the cross-connection
   - whether both legs belong to the same tributary
   - whether a path protection is included.

   The following values are possible:
   - 1WAY: the cross-connection consists of one leg given by the *in_aid* to the *out_aid*
   - 2WAY: the cross-connection consists of two legs given by the *in_aid* to the *out_aid* and the return leg given by the *out_aid* to the *in_aid*
   - 1WAYPPROT: the cross-connection consist of one leg (the worker leg) given by *in_aid* to *out_aid* and another leg (the protection leg) given by *in_aid2* to the *out_aid*. The path selector function is associated with the tributary of *out_aid*.

   Type: Alphanumeric value string.
   Range:
   - 1WAY
   - 2WAY (default)
   - 1WAYPPROT.
7. **`in_aid2`** – Input Access Identifier 2
   This parameter is applicable to path-protected cross-connections only and specifies the source of the protection leg.
   Thus *in_aid* and *out_aid* specify the working leg while *in_aid2* and *out_aid* specify the protection leg of a path-protected cross-connection. For non path-protected cross-connection the parameter *in_aid2* must be omitted.
   Both input aids (*in_aid* and *in_aid2*) have to be related to ports with the same interface mode being either SDH or SONET. This is due to different switch criteria for SONET and SDH.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
   STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
   Range: No use of the **`all`** value within the *in_aid2* is allowed.
   Default: None (a valid value is needed on case of a path-protected cross-connection – no default is defined).

Output format

If the **`DLT-CRS`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

Cross-connection can be deleted via the **DLT-CRS** command when *xconcls* = TRADITIONAL.

Deletion of 2WAY cross-connections

Please note that the 2 legs of a 2-way cross-connection can be deleted either via a single DLT-CRS command (with *cct*=2WAY) or via 2 separate DLT-CRS commands (each with *cct*=1WAY). In the latter case an additional *repleg* status change will be notified via an ED-CRS notification.

When deleting a cross-connection the cross-connection entity will be deleted as well. This deletion is notified via a DLT-CRS notification.

Updated affected entities: General

In case of deleting a path-protected cross-connection the command deletes in addition the entity of type protection group being related to that path-protected cross-connection. This deletion is notified via a DLT-PROTN-GRP notification. Please refer to the “DLT-PROTN-GRP notification” (p. 3-243) section for details.

In case the *repleg* state is changed by deleting a cross-connection an ED-CRS notification is used to notify the updated *repleg* value.

Updated affected entities: 1-way cross-connection, when not part of 2 way cross-connection

In case a 1-way cross-connection is deleted where no ″equivalent return-leg″ (see below) is already existing the affected instance of the cross-connection is deleted. This is reported by means of an DLT-CRS notification.

An ″equivalent return-leg″ means: for a 1-way cross-connection with rate N from in_trib to out_trib a return-leg would be given by

- a 1-way cross-connection from out_trib to in_trib
- with the same rate N
- and in (case of BLSR only) the same *loca* and *locz* values.

Example:

DLT-CRS-vc3:NameOfNE:1-1-#-#-6-1-1,1-1-#-#-1-1-1:88::1way;

REPT DBCHG

DLT-CRS-VC3:1-1-#-#-6-1-1,1-1-#-#-1-1-1:1WAY;

Updated affected entities: 1-way cross-connection, when part of a 2-way cross-connection

In case a 1-way cross-connection is deleted where an ″equivalent return-leg″ (see below) exists the instance of type cross-connection, which represents the leg as specified in the **`DLT-CRS`** command is deleted.

This is reported by means of an DLT-CRS notification.

Furthermore the altered *repleg* status is reported via an ED-CRS notification.

An ″equivalent return-leg″ means: when entering a 1-way cross-connection with rate N from in_trib to out_trib a return-leg would be given by

- a 1-way cross-connection from out_trib to in_trib
- with the same rate N
- and in (case of BLSR only) the same *loca* and *locz* values.

Example (Precondition: 2-way cross-connection:1-1-#-#-7-3-1 ↔ 1-1-#-#-3-4-1):

DLT-CRS-vc4:NEname:1-1-#-#-7-3-1,1-1-#-#-3-4-1:587::1way;

`REPT DBCHG`

DLT-CRS-VC4:1-1-#-#-7-3-1,1-1-#-#-3-4-1:1WAY;

`REPT DBCHG`

ED-CRS-VC4:1-1-#-#-3-4-1,1-1-#-#-7-3-1::repleg=1WAY;

Updated affected entities: 2-way cross-connection

In case a 2-way cross-connection is deleted its related instance of type cross-connection will be deleted as well.

This is reported by means of an DLT-CRS notification.

Example:

DLT-CRS-vc3:NameOfNE:1-1-#-#-6-1-1,1-1-#-#-1-1-1:88::2way;

`REPT DBCHG`

DLT-CRS-VC3:1-1-#-#-6-1-1,1-1-#-#-1-1-1:2WAY;

Updated affected entities: path-protected cross-connection

In case a path-protected cross-connection is deleted the related instance of type cross-connection will be deleted as well. This is reported by means of one DLT-CRS notification where *in_aid*, *in_aid2* can be chosen as in the delete request or flipped.

Furthermore the related instance of protection group will be deleted as well.

This is reported by means of an DLT-PROTN-GRP notification.

For details on protection group please refer to the protection group section.

Example:

DLT-CRS-vc3:NameOfNE:1-1-#-#-1-1-1,1-1-#-#-15-1-1:92::1WAYPPROT,1-1-#-#-14-1-1;

REPT DBCHG

DLT-PROTN-GRP:1-1-#-#-15-1-1:PATHDRI;

REPT DBCHG

DLT-CRS-VC3:1-1-#-#-1-1-1,1-1-#-#-15-1-1:1WAYPPROT,1-1-#-#-14-1-1;

Identification of VCG tributaries

The system shall allow to delete cross-connections at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

Specific behavior of low-order cross-connection deletion - Auto-desubstructuring of high-order tributaries

If the last low-order cross-connection is going to be removed from auto-substructured low-order entities (i.e. related to a HO-NIM entity with *substr* = YES or a HO PTF located on TransMUX) then the whole substructure is deleted:

- any performance monitoring point related to those LONIM entities,
- all LONIM entities,
- the related PTF entity.

The deletion of the latter two items are notified (by DLT-LONIM and DLT-PTF notifications).

In addition the HO NIM shall indicate that it is not substructured by setting the parameter substr to NO which is notified by a ED-rr notificiation.

In the case of TransMUX, only the LOXC related entities will be deleted. The LO path terminations (TMSVT) and the related PTF (TMSSTS1) remain unchanged.

None of the deleted settings are longer maintained. A new cross-connection will cause a creation of new substructure with default values.

Specific behavior of conversion cross-connection deletion - Auto-desubstructuring of high-order tributaries

If the last conversion cross-connection is going to be removed from auto-substructured low-order entities (i.e. related HO-NIM entity has *substr*==CNVSUB) then the whole substructure is deleted:

- any performance monitoring point related to those LONIM entities,
- all LONIM entities,
- the related PTF entity.

The deletion of the latter two items are notified (by DLT-LONIM and DLT-PTF notifications).

In addition the HO tributary shall indicate that it is not substructured by setting the parameter *substr* to NO which is notified by a ED-VC3 notificiation.

Specific behavior of conversion cross-connection deletion - Auto-configuration of the HO NIM/PTF entity related to the HO side

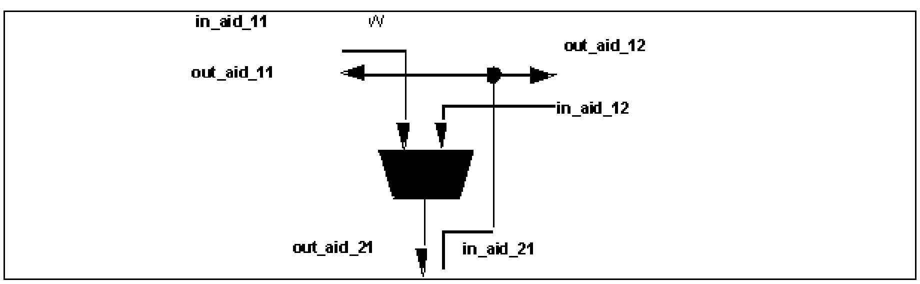
When the last conversion cross-connection is removed from a higher-order tributary, then the following parameter values of the HO NIM or HO PTF is automatically configured:

- *substr* is set to NO

This are notified with a ED-STS1 or ED-PTF notification.

Example input/output

Suppose the following compound cross-connection as depicted below is deleted: This would require the following DLT-CRS commands:




```
DLT-CRS-VC4:UNITE-NE:in_aid_11,out_aid_21:123456::1WAYPROT,in_aid_12;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;

DLT-CRS-VC4:UNITE-NE:in_aid_21,out_aid_12:123457::1way;
```

```
   UNITE-NE 00-11-14 12:01:00
M  123457 COMPLD
;
```

```
DLT-CRS-VC4:UNITE-NE:in_aid_21,out_aid_11:123458::1way;
```

```
   UNITE-NE 00-11-14 12:02:00
M  123458 COMPLD
;
```

The first command will delete the path-protected cross-connection, while the following two commands delete the bridge.

Error responses

Validation rule: Valid topology

Cross-connection command processing verifies for higher order cross-connections that the value of the topology parameter is consistent with the atomic cross-connection topology and the leg status of the existing cross-connection in accordance with the following table. Failure to meet this criteria (Validation Response = Deny) results in command rejection.

Error code: `SNVS – Status, Not in Valid State, Invalid topology`

Topology in Remove cross-connection Command (TL1 parameter: *cct*)	Topology of Existing Atomic cross-connection (internal parameter: ″Atomic cross-connection topology″ as described in the RTRV-CRS section in the *repleg* context)	Leg Status of Existing Atomic cross-connection (internal parameter: ″Leg status″ as described in the RTRV-CRS section in the *repleg* context)	Validation Response (″Accept″ means that the command is replied by a COMPLD, ″Deny″ means, that the command is rejected with the Error Code as stated above)
1WAY	point-to-point	1-way	Accept
	point-to-point	2-way	Accept
	path-protected	1-way	Deny
2WAY	point-to-point	1-way	Deny
	point-to-point	2-way	Accept
	path-protected	1-way	Deny
1WAYPPROT	point-to-point	1-way	Deny
	point-to-point	2-way	Deny
	path-protected	1-way	Accept

This table should be understood as follows:

- 1. line: suppose a 1-way cross-connection exists with a point-to-point topology – then this cross-connection can be deleted with the **`DLT-CRS`** command including the input parameter *cct*=1WAY i.e. the command will be accepted and executed;
- 2. line: suppose a 2-way cross-connection exists with a point-to-point topology – then a DLT-CRS with *cct*=1WAY will be accepted and one leg will be deleted (the other in the return direction will still exist);
- 3. line: suppose a path-protected cross-connection exists which (also) has a leg status 1-WAY – then a DLT-CRS with *cct=1way* will be denied with the Error Code `SNVS – Status, Not in Valid State, Invalid topology`; to delete this cross-connection properly the last line has to be considered, where the *cct* is compliant with the atomic cross-connection topology and the leg status.

Validation rule: Entities (tributaries) must exist

Cross-connection command processing verifies the logical tributaries specified in the command do exist.

Error code:

`IENE – Input Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Entity (cross-connection) must exist

Cross-connection command processing verifies that the specified cross-connection exists.

Error code:

`IENE – Input Entity Not Exists, Specified cross-connection does not exist`

Validation rule: LO cross-connection with specified topology must exist

Cross-connection command processing shall verify for lower order cross-connections that a lower order cross-connection with the specified topology exists.

Error code:

`IENE – Input, Entity Not Exists, Specified cross-connection does not exist`

Validation rule: Conversion cross-connection with specified topology must exist

Cross-connection command processing shall verify for conversion cross-connections that the conversion cross-connection exists and that the specified topology (parameter *cct*) is valid.

Error code:

`IENE – Input, Entity Not Exists, Specified crossconnection does not exist`

Validation rule: Valid rate

Cross-connection command processing verifies that the specified rate is equivalent to the rate of the existing cross-connection. Hereby the SONET-rates are treated identical to the SDH-rates (i.e. sts1/vc3, STS3/vc4 etc.)

Error code:

`IDNC – Input Data Not Consistent, Rate is inconsistent with existing cross-connection`

Validation rule: No cross-connection loopback existing

Cross-connection command processing denies commands requesting a cross-connection to a tributary with an existing cross-connection loopback.

Error code:

`SNVS – Status, Not in Valid State, Tributary has existing loopback`

Note: As in the **`ENT-CRS`** command there is no command validation concerning facility loopback i.e. the cross-connection deletion will be possible even with a facility loopback present.

Validation rule: No test access session

Cross-connection command processing shall deny commands requesting a cross-connection to a tributary which is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: Cross-connection must not be of *xconcls* NN

Cross-connection command processing shall verify, that the cross-connection is not of xconcls NN.

Error code:

`SNVS – Status, Not in Valid State, Cross-connection must not be of xconcls NN`

Related TL1 messages

- "ED-CRS: Edit cross-connection" (p. 3-292)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)
- "ENT-ROLL: Enter roll" (p. 3-1796)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "ED-NNCRS: Edit NN cross-connection" (p. 3-457)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)

References

GR-199-CORE

DLT-CRS-FECOM: Delete transparent DCC cross-connection

The DLT-CRS-FECOM can be initiated by the users to delete a transparent DCC cross-connection.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 3

Abortable: No.

Input format

Command syntax: **DLT-CRS-FECOM**:*tid:port_aid1,port_aid2,dcctype:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **port_aid1** – DCC port 1
   This parameter indicates one (first) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a Transparent DCC Cross Connection.
   Type: AID
   Range: SPortAID. The keyword **all** is allowed not.
4. **port_aid2** – DCC port 2
   This parameter indicates one (second) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a Transparent DCC Cross Connection.
   Type: AID
   Range: SPortAID. The keyword **all** is not allowed in the aid.
5. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS Transparent DCC Cross Connection parameters of the OP***n*** port is reported by the command.
   Type: Alphanumeric value set Range – LINE – SECTION
   Range: LINE, SECTION

Output format

In response to a valid DLT-CRS-FECOM command, the following output is returned:

```
    sid date time
 M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **DLT-CRS-FECOM** command deletes a transparent DCC cross-connection specified by *port_aid1*, *port_aid2* and *dcctype*.

The system supports only bidirectional transparent DCC cross-connections.

Update affected entities

The deletion of the transparent DCC cross-connection will be reported via DLT-CRS-FECOM notification reporting the deleted transparent DCC cross-connection.

The update of the dccstat parameter of the two involved ports will be reported via ED-FECOM notification for each port reporting the changed *dccstat* = DISABLED and *scn_dccstat* = DISABLED and *uni_dccstat* = DISABLED and *enni_dccstat* = DISABLED.

Example input/output

The following example shows DLT-CRS-FECOM commands there two transparent DCC cross-connections are deleted one for *dcctype* = SECTION and one for *dcctype* = LINE:

```
DLT-CRS-FECOM:NE-UNITE:1-1-#-#-1-1,1-1-#-#-2-1,LINE:12345;
    NE-UNITE 00-12-24 12:00:00
 M  123456 COMPLD
;
DLT-CRS-FECOM:NE-UNITE:1-1-#-#-1-1,1-1-#-#-2-1,SECTION:12345;
    NE-UNITE 00-12-24 12:00:00
 M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-CRS-FECOM** command.

Validation rule: Entity (transparent DCC cross-connection) must exist

The specified transparent DCC cross-connection must exist on both specified ports.

Error code:

`IENE – Input Entity Not Exists, Specified Transparent DCC Cross Connection does not exist`

Related TL1 messages

- "ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
- "RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection" (p. 3-2329)
- "RTRV-FECOM: Retrieve DCC related parameters" (p. 3-2407)
- "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)

DLT-CRS-FECOM notification

Output format

Command syntax: **DLT-CRS-FECOM**:*port_aid1*,*port_aid2*,*dcctype*

Output parameters

Command parameters

1. **port_aid1** – DCC port 1
   This parameter indicates one (first) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a Transparent DCC Cross Connection.
   Type: AID
   Range: SPortAID. The keyword **all** is allowed not.
2. **port_aid2** – DCC port 2
   This parameter indicates one (second) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a Transparent DCC Cross Connection.
   Type: AID
   Range: SPortAID. The keyword **all** is not allowed in the aid.
3. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS Transparent DCC Cross Connection parameters of the OP***n*** port is reported by the command.
   Type: Alphanumeric value set Range – LINE – SECTION
   Range: LINE, SECTION

Example output

The following example shows DLT-CRS-FECOM notifications reporting the deletion of transparent DCC cross-connections:

```
   UNITE-NE 00-12-24 12:00:00
 A  000 REPT DBCHG
   ″TIME=12:00:00,DATE=00-12-24:DLT-CRS-FECOM::1-1-#-#-1-1,1-1-#-2-1,LINE″
;

   UNITE-NE 00-12-24 12:00:00
 A  000 REPT DBCHG
   ″TIME=12:00:00,DATE=00-12-24:DLT-CRS-FECOM::1-1-#-#-1-1,1-1-#-2-1,SECTION″
;
```

DLT-CRS notification

Output format

Syntax: **DLT-CRS**-*modifier*:*in_aid*,*out_aid*:*cct*,[*in_aid2*];

Output parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **DLT-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - AU3TU3
   - TU3AU3

   No use of the *all* value is allowed.
2. **in_aid** – Input Access Identifier
   This parameter specifies the source of the cross-connection.
   In case of a 1-way cross-connection the source of a cross-connection is unambiguous. In this case the cross-connection will be retrieved by the RTRV-CRS, which has the identical *in_aid*.
   In case of a 2-way cross-connection the source is not defined (the source could be the *in_aid* or the *out_aid* of the cross-connection). In this case both aids (*in_aid* or *out-aid*) of the cross-connection can be used as the *in_aid* parameter of the **RTRV-CRS** command. The **RTRV-CRS** command will then return the *in_aid* –

parameter used in the **RTRV-CRS** command as the *in_aid* of the RTRV-CRS reply.
Example:
ENT-CRS-VC4:NEname:1-1-#-#-1-1-1,1-1-#-#-2-1-1:123456:2way;
RTRV-CRS-VC4:NEname:1-1-#-#-2-1-1:123457;
M 123457 COMPLD ″1-1-#-#-1-1-,1-1-#-#-2-1-1:vc4….. ;
Type: GenTribAID.
Range: The **all**-parameter is accepted for the slot-, port- and trib-part of the AID. For the port the all-parameter is to be distinguished between **all** when referring OP***n*** packs and GE***n*** **vall** when referring the GE***n***.

For example the following *in_aids* are valid:

- 1-1-#-#-all-all-all specifying all tributaries and all ports in the whole shelf
- 1-1-#-#-2-1-all specifying all tributaries of the first port of the pack in slot-number=02
- 1-1-#-#-3-vall-all specifying all tributaries of the first port of the (GE***n***) pack in slot-number=03

3. **out_aid** – Output Identifier
This parameter specifies the tributary which is the sink of the cross-connection.
Type: STS1NumAID (when using OP***n*** packs) or VCGNumAID (when using GE***n***).
Range: n.a. (**all** values shall never occur as part of an output parameter).

4. **cct** – cross-connection Topology
This parameter specifies the topology of the cross-connection to be established via the **ENT-CRS** command.
The topology defines
   - the number of legs of the cross-connection
   - whether both legs belong to the same tributary
   - whether a path protection is included.

   The following values are possible:
   - 1WAY: the cross-connection consists of one leg given by the *in_aid* to the *out_aid*
   - 2WAY: the cross-connection consists of two legs given by the *in_aid* to the *out_aid* and the return leg given by the *out_aid* to the *in_aid*
   - 1WAYPPROT: the cross-connection consist of one leg (the worker leg) given by *in_aid* to *out_aid* and another leg (the protection leg) given by *in_aid2* to the *out_aid*. The path selector function is associated with the tributary of *out_aid*.

   Type: Alphanumeric value string.
   Range:
   - 1WAY
   - 2WAY (default)
   - 1WAYPPROT.

5. **`in_aid2`** – Input Access Identifier 2
   This parameter is applicable to and is present for path-protected cross-connections only and specifies either the source of the protection leg or the source of the working leg.
   This depends on the value of the *repleg* parameter:
   - E.g. for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSW the worker leg is given by IN1 -> OUT and the protection leg is given by IN2 -> OUT;
   - for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSP the worker leg is given by IN2 -> OUT and the protection leg is given by IN1 -> OUT.

   Thus *in_aid* and *out_aid* specify one leg of the legpair of a path-protected cross-connection, while *in_aid2* and *out_aid* specify the second leg.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
   STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
   Note: Using VCGNumAID (when using e.g. GE***n*** packs) is NOT applicable as those packs can not be the input to a path-protected cross-connection. Therefore specific AID types are mentioned instead of GenTribAID.
   Range: n.a. (**`all`** and **`usall`** values shall never occur as part of an output parameter).

Example output

Refer to the ″Detailed behavior description″ in “DLT-CRS: Delete cross-connection” (p. 3-145).

DLT-EC1 notification

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: EC1 port AID

Example Output

```
   UNITE-NE 04-12-23 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=04-12-23:DLT-EC1:1-1-#-#-21-1”
;
```

DLT-ENNIMANROUTE: Delete manual route from the E-NNI node

The **DLT-ENNIMANROUTE** command is used to delete manual routes, previously entered via ENT-ENNIMANROUTE, from the E-NNI node.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-ENNIMANROUTE**-modifier*:tid::ctag:::[spec_block];*

Input parameters

Command parameters

1. **modifier**
The parameter modifier indicates whether the manual route is used to reach a signaling controller (SC) or a routing controller (RC) subnet.
Type: Alphanumeric value set.
Range: SC or RC
2. **tid** – Target Identifier
See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – Correlation Tag
This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
Type: Alphanumeric string
Range: non-zero decimal number or identifier of up to 6 characters.
Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **neighborSIGPCIPAddress** – Neighbor Signaling Controller IP Address
This parameter is used to indicate the IP address of a neighboring signaling controller that is connected via a LAN interface.
The term "neighbor" refers to logical neighborship. The neighboring SC is not necessarily a physical neighbor.
The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.

Type: Four dot separated numeric strings.
Range: Each numeric string: 0 .. 255.
The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.

2. **`neighborSIGPCID`** – Neighbor Signaling Controller Identifier
   This parameter is used to indicate the identifier of a neighboring signaling controller that is connected via a LAN interface.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
3. **`neighborRCIPAddress`** – Neighbor Route Controller IP Address
   This parameter is used to indicate the IP address of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
4. **`neighborRCIPSubnetMask`** – Neighbor Route Controller IP Subnet Mask
   This parameter is used to indicate the subnet mask of the IP subnet of a neighboring route controller.
   The IP subnet that can be reached via the manual route is described by the tuple neighborRCIPAddress + neighborRCIPSubnetMask.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
5. **`if_id`** – interface id
   This parameter is used to indicate the interface associated with the manual route.
   Type: AID
   Range: SPortAID or LanPort AID (the keyword “all” is not used)
6. **`dcctype`** – DCC type
   This parameter is used to further specify the DCC type, in case the *if_id* refers to a DCC channel.
   Type: Alphanumeric value set.
   Range: LINE or SECTION
   Optional parameter: This parameter must be present if the interface *if_id* is a DCC. It must not be present, if the interface *if_id* is no DCC.

Output format

If the DLT-ENNIMANROUTE is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the “RTRV-HDR: Retrieve header” (p. 3-2417) command.

Detailed behavior description

The **DLT-ENNIMANROUTE** command shall delete a manual route entry from the E-NNI node. The deletion of the manual route is notified with DLT-ENNIMANROUTE Notifications.

Example input/output

The following example shows the result of a DLT-ENNIMANROUTE command:

```
DLT-ENNIMANROUTE-SC:LT-UNITE::123456::: neighborSIGPCIPAddress=30.24.15.10,
      neighborSIGPCID=0.0.0.42, if_id=1-1-lan2;

   LT-UNITE 06-10-09 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-ENNIMANROUTE** command.

Validation rule: Parameter Combinations

The command is rejected if the combination of modifier and parameters is not one of the following:

- modifier==SC and interface type is point-to-multipoint and list of parameters is: *neighborSIGPCIPAddress*, *neighborSIGPCID*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a LAN and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *lan_nh_ifipad*, *if_id*, where *if_id* refers to a LAN interface
- modifier==RC and interface is a point-to-point interface and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, *dcctype*, where *if_id* refers to a DCC

Error code:

`IDNV – Input, Data Not Valid, Combination of Modifier/Parameters not allowed`

Note that the parameter *lan_nh_ifipad* is by intention not part of the command, as it is not allowed to enter manual routes to the same destination via the same LAN but via different gateways. Therefore the *lan_nh_ifipad* is irrelevant for deleting an entry.

Validation rule: Port not used for ENNI

If the *if_id* and *dcctype* (in case of a DCC) is not in use for ENNI, i.e. is not assigned to the enninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, port not in use for ENNI`

Validation rule: Manual Route entry must exist

If the given manual route does not exist, the command is rejected.

A manual route exists if one of the following conditions is fulfilled:

- an entry with identical *neighborSIGPCIPAddress*, *neighborSIGPCID*, *lan_nh_ifipad* and *if_id* is already provisioned
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *lan_nh_ifipad*, and *if_id* is already provisioned, where *if_id* refers to a LAN interface
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, and *dcctype* is already provisioned, where *if_id* refers to a DCC

Error code:

`IDNV – Input, Data Not Valid, Manual route entry does not exist`

Related TL1 messages

- "RTRV-ENNIMANROUTE: Retrieve all manual routes of the E-NNI node" (p. 3-2348)
- "DLT-ENNIMANROUTE: Delete manual route from the E-NNI node" (p. 3-164)

DLT-ENNIMANROUTE notification

Output format

The DLT-ENNIMANROUTE notification has the following specific syntax:

`DLT-ENNIMANROUTE`-modifier*:::spec_block*

Output parameters

Command parameters

1. **`modifier`**
   The **`DLT-ENNIMANROUTE`** command is used on ASON domain border NEs, i.e. NEs that have ENNI interfaces to other ASON domains, to enter (add) manual routes to other ASON domains into the forwarding table of the E-NNI node.

   There are two types of manual routes:

   - manual routes to neighboring signaling controllers
   - manual routes to neighboring routing controllers

   The command modifier is used to distinguish these two types of manual routes.
   Note that this command cannot be used to enter manual routes to neighboring signaling controllers that are connected via point-to-point interfaces, as these manual routes are implicitely added upon port provisioning.
   Refer to the Example Input/Output section for provisioning examples.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **`DLT-ENNIMANROUTE`**-modifier*:tid::ctag:::[spec_block];*

Input parameters

Command parameters

1. **`modifier`**
   The parameter modifier indicates whether the manual route is used to reach a signaling controller (SC) or a routing controller (RC) subnet.
   Type: Alphanumeric value set.
   Range: SC or RC
2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`neighborSIGPCIPAddress`** – Neighbor Signaling Controller IP Address
   This parameter is used to indicate the IP address of a neighboring signaling controller that is connected via a LAN interface.
   The term "neighbor" refers to logical neighborship. The neighboring SC is not necessarily a physical neighbor.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
2. **`neighborSIGPCID`** – Neighbor Signaling Controller Identifier
   This parameter is used to indicate the identifier of a neighboring signaling controller that is connected via a LAN interface.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.

3. **`neighborRCIPAddress`** – Neighbor Route Controller IP Address
   This parameter is used to indicate the IP address of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
4. **`neighborRCIPSubnetMask`** – Neighbor Route Controller IP Subnet Mask
   This parameter is used to indicate the subnet mask of the IP subnet of a neighboring route controller.
   The IP subnet that can be reached via the manual route is described by the tuple neighborRCIPAddress + neighborRCIPSubnetMask.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
5. **`lan_nh_ifipad`** – Next Hop Interface IP Address for LAN Interfaces
   This parameter is used to indicate the interface IP address of the next hop of a manual route using a LAN. This parameter is insignificant for point-to-point interfaces.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the interface referred by *if_id* is a LAN.
6. **`if_id`** – interface id
   This parameter is used to indicate the interface associated with the manual route.
   Type: AID
   Range: SPortAID or LanPort AID (the keyword “all” is not used)
7. **`dcctype`** – DCC type
   This parameter is used to further specify the DCC type, in case the *if_id* refers to a DCC channel.
   Type: Alphanumeric value set.
   Range: LINE or SECTION
   Optional parameter: This parameter must be present if the interface *if_id* is a DCC. It must not be present, if the interface *if_id* is no DCC.

Output format

If the **`DLT-ENNIMANROUTE`** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Detailed behavior description

The **DLT-ENNIMANROUTE** command shall add a manual route entry to the E-NNI node. The update of the manual routes are notified with DLT-ENNIMANROUTE Notifications.

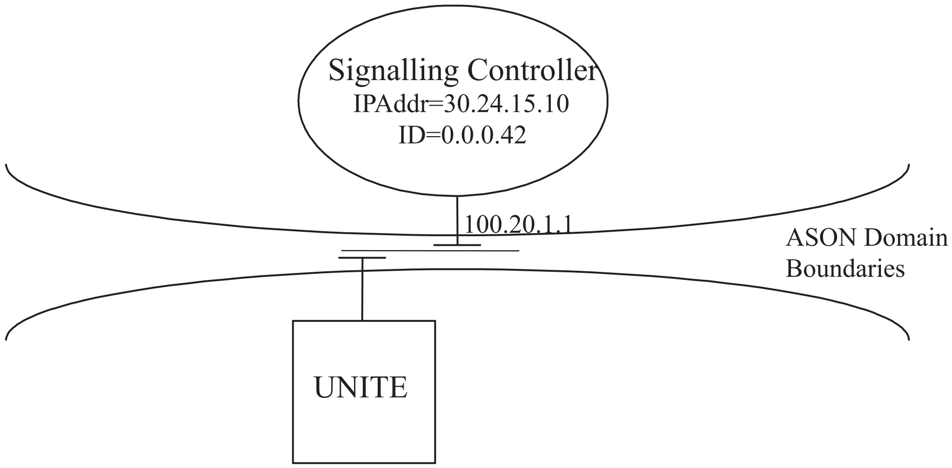
Example input/output

```
ENT-ENNIMANROUTE-RC:LT-UNITE::123456:::neighborRCIPAddress=95.22.10.1,
      neighborRCIPSubnetMask=255.255.255.0, lan_nh_ifipad=100.20.1.1, if_id=1-1-
      lan2;

   LT-UNITE 06-10-09 08:00:00
M  123456 COMPLD
;
```

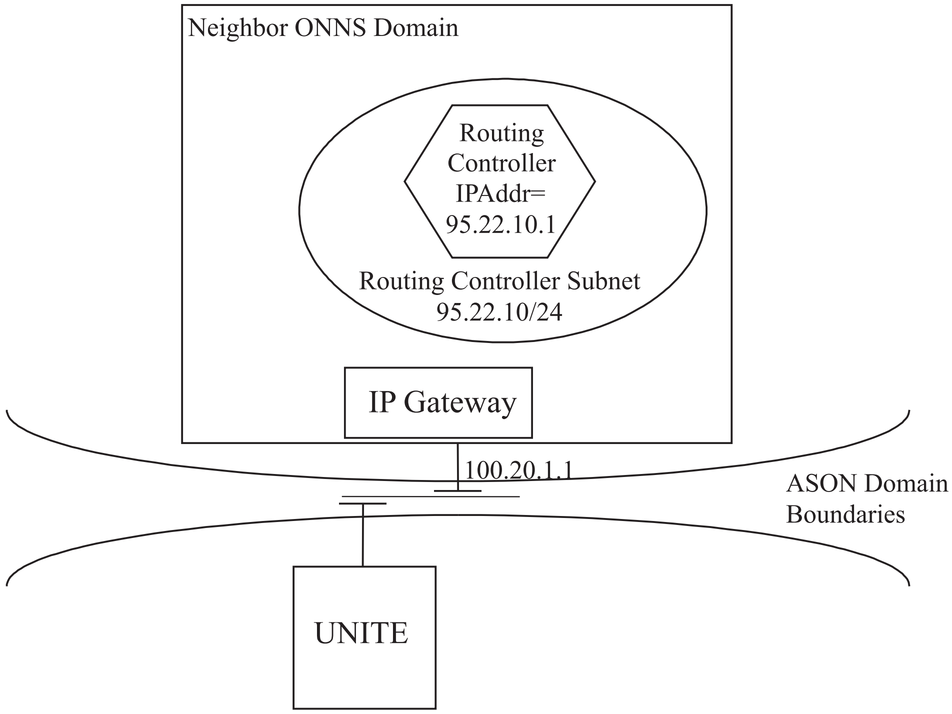
Signaling Controller Provisioning Example

```
ENT-ENNIMANROUTE-SC:LT-
UNITE::123456:::neighborSIGPCIPAddress=30.24.15.10,
neighborSIGPCID=0.0.0.42, lan_nh_ifipad=100.20.1.1,
if_id=1-1-lan2;
```



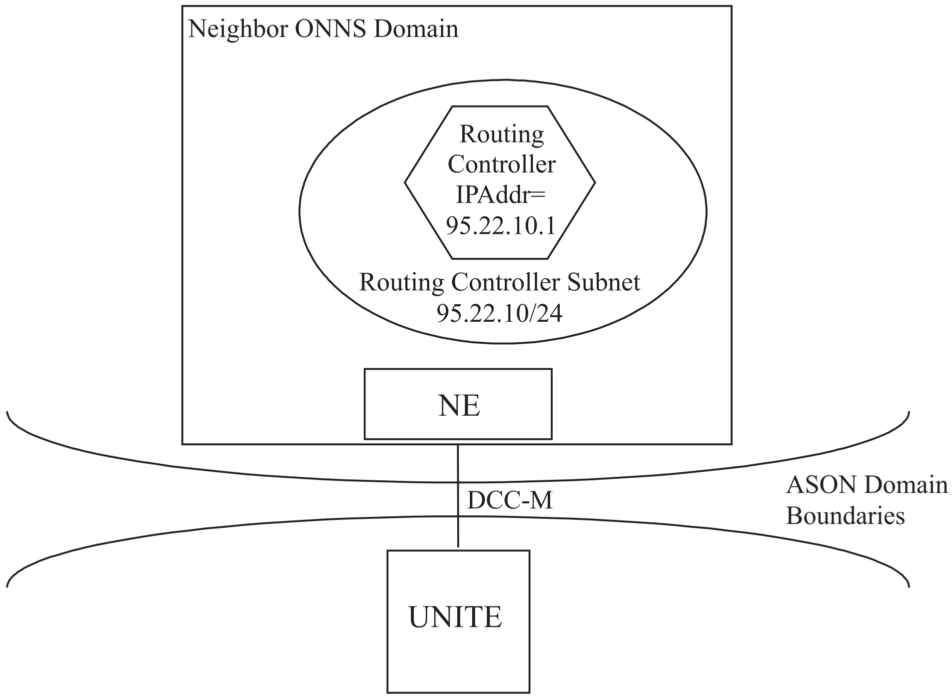
Routing Controller Provisioning Example P2MP

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0,
lan_nh_ifipad=100.20.1.1, if_id=1-1-lan2;
```



Routing Controller Provisioning Example P2P

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0, if_id=1-1-#-#-5-3,
dcctype=LINE;
```



Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **DLT-ENNIMANROUTE** command.

Validation rule: Parameter Combinations

The command is rejected if the combination of modifier and parameters is not one of the following:

- modifier==SC and if is a LAN and list of parameters is: *neighborSIGPCIPAddress*, *neighborSIGPCID*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a LAN and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a point-to-point interface and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, *dcctype*

Error code:

`IDNV – Input, Data Not Valid, Combination of Modifier/Parameters not allowed`

Validation rule: Not allowed to use LAN1 for E-NNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for ENNI`

Validation rule: LAN port not used for E-NNI

If the LAN port specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LAN port not in use for ENNI`

Validation rule: DCC port not used for E-NNI

If the DCC specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, DCC not in use for ENNI`

Validation rule: No duplicate Manual Route entries

The command is denied, if a duplicate manual route entry is entered.

A manual route is considered to be duplicate if one of the following conditions is fulfilled:

- an entry with identical *neighborSIGPCIPAddress*, *neighborSIGPCID*, and *if_id* is already provisioned
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, and *if_id* is already provisioned, where *if_id* refers to a LAN interface
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, and *dcctype* is already provisioned, where *if_id* refers to a DCC

Error code:

`IDNV – Input, Data Not Valid, Manual route entry already exists`

Note that the *lan_nh_ifipad* is by intention not considered, as it is not allowed to have more than one gateway to one destination on one and the same LAN.

Validation rule: Check if limit of 20 Manual Routes per LAN port exceeded

The addition of a manual route entry for a LAN port is denied, if already 20 manual route entries exist for the LAN port.

Error code:

`SSRE – Status, System Resources Exceeded, limit of manual route entries exceeded`

Validation rule: Check if limit of 64 Manual Routes per ENNI Node exceeded

The addition of a manual route entry is denied, if already 64 manual route entries exist for the ENNI node.

Error code:

`SSRE – Status, System Resources Exceeded, ENNI limit of manual route entries exceeded`

This requirement is independent from the type of interface (LAN or point-to-point).

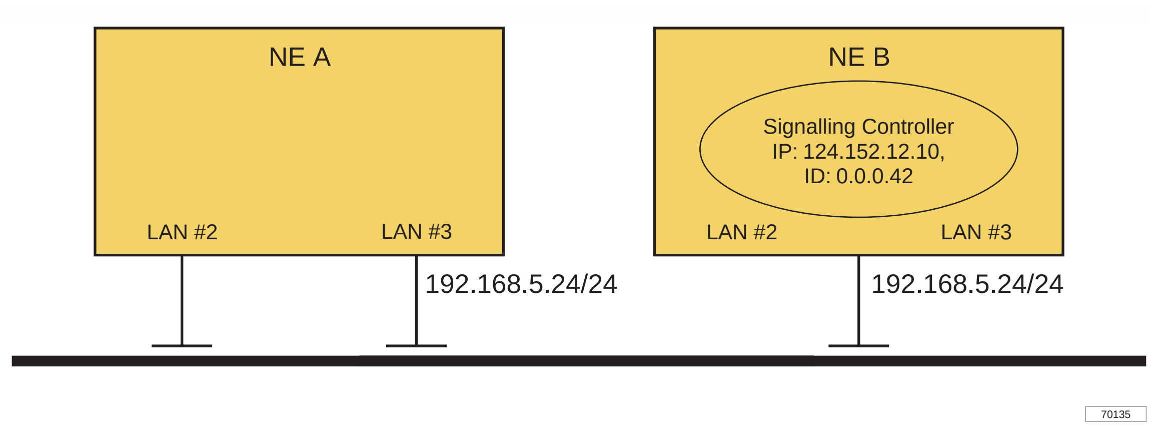
Validation rule: No more than one LAN connected to same next hop

The command is rejected if the provided *lan_nh_ifipad* is already provisioned for any other LAN interface.

Error code:

`IDNV – Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`

This requirement prevents a configuration as shown below, which is desirable, but currently not supported:



Related TL1 messages

- “RTRV-MANROUTE-LAN: Retrieve manual route at LAN port” (p. 3-2516)
- “DLT-MANROUTE-LAN: Delete manual route at LAN port” (p. 3-203)

DLT-EPORT notification

Output format

The **DLT-EPORT** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **DLT-EPORT** notification has the following specific syntax:

DLT-EPORT:*aid;*

Output parameters

Command parameters

aid – Access Identifier

The access identifier specifies the Ethernet port to be operated on.

Type: AID

Range: EPortAID as defined in AID structure. The usage of the following values are excluded:

- *all* in the BayAID definition
- *all* in the Simple_ShelfAID definition
- *usall* in the Simple_UnivSlotAID definition
- *all* in the Simple_UnivSlotAID definition
- *all* in the Simple_EPortAID definition.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:DLT-EPORT:1-1-#-#-1-1:::”
;
```

DLT-EQPT (slot): Delete equipment (slot)

The **DLT-EQPT** command deletes equipment that has been previously created with the **ENT-EQPT** command or via CP plug-in (auto-provisioning).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-EQPT**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The CTL cannot be deprovisioned.
   The AID of a slot.
   Type: AID
   Range:
   - SlotAID
   - UnivSlotAID
   - XCSlotAID
   - CTLSlotAID.

   The keyword **usall** is allowed for Simple_UnivSlotAID.
   The keyword **xcall** is allowed for Simple_XCSlotAID.
   The keyword **ctlall** is allowed for Simple_CTLSlotAID.
   The keyword **all** is allowed for Simple_SlotAID.
   The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **DLT-EQPT** command completes successfully, then the following normal completion response is returned:

```
sid date time
```

```
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command deletes a circuit pack for a specified slot.

The deletion is reported by means of an DLT-EQPT notification for circuit pack.

For de-provisioning of a slot some conditions must be fulfilled.

Behavior of *actlname* during deprovisioning

When a circuit pack is deleted from the specified slot, then the actlname of the slot is modified and contain all allowed functional names for provisioning, according to the current provisioning status of the system.

Behavior of deprovisioning for multislot circuit packs

The attribute *actlname* of the neighbor slots which were covered by the multislot circuit pack (e.g. OP40, LOXC40G2S, LOXC40G3S) is modified and set from empty to the list of accepted packs.

The attributes *nvmname*, *nvmqual* of the neighbor slots which were covered by the multislot circuit pack (e.g. OP40, LOXC40G2S, LOXC40G3S) are modified and set to UNDEFINED, UNDEF.

The update is reported by means of an **`ED-EQPT`** notification for this slot.

Update affected entities

The command deletes in addition a set of entities of different types depending on the provisioned circuit pack

1. One instance of type circuit pack is deleted with attribute *aid*. This is reported by means of DLT-EQPT notification for the CP.
2. The instances of port STM-16 is deleted if *nvmname* of the addressed circuit pack equals *OP2G5* and the port attribute *istd* equals *SDH*. This is reported by means of DLT-STM16 notification of port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.
3. The instances of port OC-48 are deleted if *nvmname* of the addressed circuit pack equals *OP2G5* and the port attribute *istd* equals *SONET*. This is reported by means of DLT-OC48 notification of port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.

4. One instance of port STM-64 is deleted if *nvmname* of the addressed circuit pack equals *OP10* and the port attribute *istd* equals 'SDH'. This is reported by means of DLT-STM64 notification of port.
5. One instances of port OC-192 is deleted if *nvmname* of the addressed circuit pack equals *OP10* and the port attribute *istd* equals *SONET*. This is reported by means of DLT-OC192 notification of port.
6. One instance of port STM-256 is deleted if *nvmname* of the addressed circuit pack equals *OP40* and the port attribute *istd* equals *SDH*. This is reported by means of DLT-STM256 notification of port.
7. One instances of port OC-768 is deleted if *nvmname* of the addressed circuit pack equals *OP40* and the port attribute *istd* equals *SONET*. This is reported by means of DLT-OC768 notification of port.
8. 48, 192, 768 VC3 tributaries are deleted for each existing SDH port. This is reported by means of DLT-VC3 notifications for tributary.
9. 48, 192, 768 STS1 tributaries are deleted for each existing SONET port. This is reported by means of DLT-STS1 notifications for tributary.
10. The dcc channel entities are deleted for each existing SDH/SONET port. This is reported by means of DLT-FECOM notification for dcc channel.
11. One instance of 1+1 equipment protection group is deleted if nvmname of the addressed circuit pack equals 'to one out of {XC160, XC32 – , XC640} (not all variants are supported in all releases) and another XC circuit pack with the same nvmname exists. This are reported by means of DLT-PROTNGRP notification for the 1+1 protection group.
12. The parameters *nvmname, nvmqual* of the slot will be set to UNDEFINED, UNDEF, which will be reported by an ED-EQPT notification for slot.

Update affected entities - GE1

In addition to the rules listed above the command deletes the following entities if *nvmname* equals GE1:

- all instances of the entity EPORT owned by this circuit pack are deleted. This is reported by means of DLT-EPORT notification
- all instances of the entity VCG owned by this circuit pack are deleted. This is reported by means of DLT-VCG notification.
- For each VCG to be deleted all tributaries owned by this VCG are deleted. This is reported by means of DLT-VCGTRIB notification.

Update affected entities - Port Entities of OP622

In addition to the rules listed above the command deletes the following entities if *nvmname* equals OP622:

- 12 VC3 tributaries are deleted for each existing SDH ports. This is reported by means of DLT-VC3 notifications for tributary.
- 12 STS1 tributaries are deleted for each existing SONET port. This is reported by means of DLT-STS1 notifications for tributary.
- The instances of port STM-4 are deleted if the port attribute *istd* of the node equals ’SDH’. This is reported by means of DLT-STM4 notification for port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.
- The instances of port OC-12 are deleted if the port attribute *istd* of the node equals ’SONET’. This is reported by means of DLT-OC12 notification for port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.

Update affected entities - Port Entities of OP155

In addition to the rules listed above the command deletes the following entities if *nvmname* equals OP155:

- 3 VC3 tributaries are deleted for each existing SDH ports. This is reported by means of DLT-VC3 notifications for tributary.
- 3 STS1 tributaries are deleted for each existing SONET port. This is reported by means of DLT-STS1 notifications for tributary.
- The instances of port STM-1 are deleted if the port attribute *istd* of the node equals SDH. This will be reported by means of DLT-STM1 notification for port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.
- The instances of port OC-3 are deleted if the port attribute *istd* of the node equals SONET. This will be reported by means of DLT-OC3 notification for port. The number of port instances to be deleted depends on the capabilities of the actual circuit pack.

Update affected entities - Port Entities of EP155

In addition to the rules listed above the command shall delete the following entities if *nvmname* equals EP155:

- All tributaries are deleted for each existing SDH ports. This are reported by means of DLT-VC3 notifications for tributary.
- 8 STM-1E port instances of the actual circuit pack is deleted. This are reported by means of DLT-STM1E notification for port.

Update affected entities - GE1-VRTSW Entity

If nvmname is GE1 the command has the following additional side effects:

- If *vlantgmd* is 802.1tag the command shall delete the instance of entity *vrtsw*. This is notified by a DLT-VRTSW notification.

Update affected entities - OPLB

The command shall delete in addition following entities:

- Some instances of type socket is deleted for each socket contained on the circuit pack one. This are reported by means of DLT-EQPT notifications for sockets.
- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

If optical modules are provisioned when the OPLB circuit pack will be de provisioned implicit also the optical modules will be de provisioned at first. The actions described in the DLT-EQPT (socket) command for de provisioning of modules will be performed in this case.

Update affected entities - OP2G5D

The command shall delete in addition following entities:

- Some instances of type socket is deleted for each socket contained on the circuit pack one. This are reported by means of DLT-EQPT notifications for sockets.
- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

If optical modules are provisioned when the OP2G5D circuit pack will be de provisioned implicit also the optical modules will be de provisioned at first. The actions described in the DLT-EQPT (socket) command for de provisioning of modules will be performed in this case.

Update affected entities - OP10D

The command shall delete in addition following entities:

- Some instances of type socket is deleted for each socket contained on the circuit pack one. This are reported by means of DLT-EQPT notifications for sockets.
- One instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

If optical modules are provisioned when the OP10D circuit pack will be de provisioned implicit also the optical modules will be de provisioned at first. The actions described in the DLT-EQPT (socket) command for de provisioning of modules will be performed in this case.

Update affected entities - OPT2G5

The command deletes in addition following entities:

- All instnces of VC-4 or STS-3c tributaries are deleted. This are reported by means of DLT-VCGTRIB notification.
- All instance of entity VCG-TRANS is deleted. This are reported by means of DLT-VCG-TRANS notification.
- All instances of type socket is deleted (one for each socket contained on the circuit pack). This are reported by means of DLT-EQPT notifications for sockets.
- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

If optical modules are provisioned when the OPT2G5 circuit pack will be de provisioned implicit also the optical modules will be de provisioned at first. The actions described in the DLT-EQPT (socket) command for de provisioning of modules will be performed in this case.

Update affected entities - LOXC

The command deletes the following entities:

- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

Update affected entities - EP51

The command deletes the following entities if nvmname equals EP51:

- 1 STS1 path termination is deleted for each existing T3 port. This are reported by means of DLT-PTF-STS notifications.
- 1 STS1 tributary is deleted for each existing EC1 port. This are reported by means of DLT-STS1 notification.
- All instances of port T3 is deleted. This are reported by means of DLT-T3 notification for port. The number of port instances to be deleted depends on the actual provisioned port types.
- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

On the EP51 pack, each port can be individually provisioned as T3 or EC1 port. Depending on the port provisioning status, one portion of the 36 ports may be of type T3 while the other ports are EC1.

Update affected entities - TransMUX

The command shall delete the following entities if *nvmname* equals TMUX2G5:

The instance of type circuit pack is deleted. This is reported by means of a **`DLT-EQPT`** notification for circuit pack.

Update affected entities - ES64U

The command shall delete the following entities if *nvmname* equals ES64U:

The instance of type circuit pack is deleted. This is reported by means of a **DLT-EQPT** notification for circuit pack.

Changes for slots allowing to carry a TransMUX pack when deprovisioning a TransMUX pack

When deprovisioning the TransMUX pack, the attribute *actlname* of slots which are allowed to carry TransMUX packs is modified. The update is reported by means of an **ED-EQPT** notification for the affected slots.

Changes for slots allowing to carry a ES64U pack when deprovisioning an ES64U pack

When deprovisioning the ES64U pack, the attribute *actlname* of slots which are allowed to carry ES64U packs is modified. The update is reported by means of an **ED-EQPT** notification for the affected slots.

Update affected entities - OP10/XTTC and OP10XTTL

The command deletes the following entities:

- 192 VC3 tributaries are deleted for the existing SDH port. This are reported by means of a DLT-VC3 group notification for tributary
- 192 STS1 tributaries are deleted for the existing SONET port. This are reported by means of a DLT-STS1 group notification for tributary.
- Two dcc channels entities are deleted (one per type section/dccr and one type line/dccm) for the existing SDH/SONET port. This are reported by means of DLT-FECOM notifications for dcc channels.
- The existing STM-64 port is deleted. This are reported by means of a DLT-STM64 notification for port.
- The existing OC-192 port is deleted. This are reported by means of a DLT-OC192 notification for port.
- The existing OTN port of type OTU2 is deleted. This are reported by means of a DLT-OTU2 notification for OTN port.
- One instance of type circuit pack is deleted. This are reported by means of a DLT-EQPT notification for circuit pack.

On the EP51 pack, each port can be individually provisioned as T3 or EC1 port. Depending on the port provisioning status, one portion of the 36 ports may be of type T3 while the other ports are EC1.

Update affected entities - GE10PL1

The command shall delete in addition following entities:

- Nine instances of type socket is deleted for each socket contained on the circuit pack one. This are reported by means of DLT-EQPT notifications for sockets.
- one instance of type circuit pack is deleted. This are reported by means of an DLT-EQPT notification for circuit pack.

If optical modules are provisioned when the GE10PL1 circuit pack will be deprovisioned then at first also the optical modules will be deprovisioned implicitly. The actions described in the DLT-EQPT (socket) command for de provisioning of modules will be performed in this case.

Group notification for STS-1/AU-3 tributaries

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

Group notification for VCG tributaries

Example: The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like: 1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping) 1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping).

Disabling of PM monitor

See Detailed behavior of the “ED-STM16: Edit STM16 port” (p. 3-1035) w.r.t PM impact as side of effect of port/tributary deletion.

Example input/output

```
DLT-EQPT:LT-UNITE:1-1-#-#-1:123456;

   LT-UNITE 00-11-14 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-EQPT** command.

Validation rule: No cross-connections allowed

Check of cross-connections involving the circuit pack to be removed.

Error code:

```
SNVS – Status, Not in Valid State, cross-connections are still existing
```

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure allowed

Check if lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure exist which involve the circuit pack to be removed. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, but no higher order cross-connections are present, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: No XC loopbacks allowed

Check of XC loopbacks involving the circuit pack to be removed.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: No protection groups allowed

Check of protection groups involving the circuit pack to be removed.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: No transparent DCC connections

Check of used transparent DCC connections involving the circuit pack to be removed.

Error code:

`SNVS – Status, Not in Valid State, transparent DCC connections are still existing`

Validation rule: Not used as timing reference

Check of that none of the ports of the involving circuit pack is used as timing reference

Error code:

`SNVS – Status, Not in Valid State, at least one port used as timing reference`

Validation rule: Not used for OW

Check of that no connection to a OW port exists.

Error code:

`SNVS – Status, Not in Valid State, connection to OW port exists`

Validation rule: CP must be plugged out

Check of that the involved circuit pack is plugged out.

Error code: `SNVS – Status, Not in Valid State, the CP is not plugged out`

Validation rule: Check whether system is maintenance condition

Preprovisioning is not allowed, when the system in maintenance condition (see "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400)).

Error code:
`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Reject Request for CTL slots

If *ctlw* or *ctlp* is specified as Simple_CTLSlotAID the command is rejected.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Check if related port are not in use by ONNS

If the provisioned circuit pack has ports with portClass = INNI or portClass = EDGE with networkTNAType=IPV4 and networkTNAAddress=0.0.0.0, the following must be fulfilled:

1. No cross-connection may exist on any tributary of the port.
2. portControl must be LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Validation rule: Check if related port are not in use by ONNS

De-provisioning is not allowed if the provisioned circuit pack has ports with portClass=UNI or portClass=eNNI or portClass = EDGE with networkTNAAddress≠default value.

Error code:
`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Validation rule: Deletion of XC circuit packs while ONNS is active

De-provisioning of worker XC pack is not allowed if the parameter xcsetup has the value ″SWISS″.

Error code: `SNVS – Status, Not in Valid State, ONNS is active`

Validation rule: Check whether LOXC pack is in use

If an attempt is made to deprovision an LOXC pack with *loswcap* not set to 0 while LO cross-connections, conversion cross-connections, cross-connection loopbacks, or LO substructure exist at one or more HO tributaries, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Check whether TransMUX pack is in use

If an attempt is made to deprovision a TransMUX pack while higher-order or lower-order cross-connections exist for the pack, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, HO or LO cross-connections still exist`

Validation rule: Check whether ES64U circuit pack is in use

If an attempt is made to deprovision an ES64U pack while higher-order cross-connections exist at the ES64U, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Related TL1 messages

- "RTRV-EQPT (slot): Retrieve equipment (slot)" (p. 3-2386)
- "ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
- "RTRV-STATE-EQPT (slot): Retrieve equipment state (slot)" (p. 3-2839)
- "ED-STATE-EQPT (system): Edit equipment state" (p. 3-866)

References

GR-199-CORE

DLT-EQPT (socket): Delete equipment (socket)

Output format

The **DLT-EQPT** command shall delete module that has been previously created with the **ENT-EQPT** command or via module plug-in (autoprovisioning).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no.

Input format

Command syntax: **DLT-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a socket.
   Type: AID
   Range: SocketAID. The keywords *all*, *usall*, *scall* is not allowed on any level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **DLT-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the “RTRV-HDR: Retrieve header” (p. 3-2417) command.

Detailed behavior description

The command shall delete a module for a specified socket. Since the creation of the module is not reported no DLT-EQPT notification for module will be reported.

Update acceptable modules list

If the command leads to a change of the Acceptable Modules List (list of actlname/actlqual), then the change is reported by means of an **`ED-EQPT`** notification.

Update affected entities

The command updates / deletes in addition a set of entities of different types depending on the provisioned optical module:

1. 3, 12, 48, 192 VC3 tributaries are deleted for each existing SDH port. This are reported by means of DLT-VC3 notifications for tributary.
2. 3, 12, 48, 192 STS1 tributaries are deleted for each existing SONET port. This are reported by means of DLT-STS1 notifications for tributary.
3. The dcc channel entities are deleted for each existing SDH/SONET port. This are reported by means of DLT-FECOM notification for dcc channel.
4. One instance of port STM-1 is deleted if *nvmname* of the addressed optical module equals *OM155* and the port attribute *istd* equals SDH. This are reported by means of DLT-STM1 notification of port.
5. One instances of port OC-3 is deleted if *nvmname* of the addressed optical module equals OM155 and the port attribute *istd* equals SONET. This are reported by means of DLT-OC3 notification of port.
6. One instance of port STM-4 is deleted if *nvmname* of the addressed optical module equals OM622 and the port attribute *istd* equals SDH. This are reported by means of DLT-STM4 notification of port.
7. One instances of port OC-12 is deleted if *nvmname* of the addressed optical module equals OM622 and the port attribute *istd* equals SONET. This are reported by means of DLT-OC12 notification of port.
8. One instance of port STM-16 is deleted if nvmname of the addressed optical module equals ’OM2G5’ and the port attribute istd equals ’SDH’. This are reported by means of DLT-STM16 notification of port.
9. One instances of port OC-48 is deleted if *nvmname* of the addressed optical module optical equals OM2G5 and the port attribute *istd* equals SONET. This are reported by means of DLT-OC48 notification of port.
10. One instance of port STM-64 is deleted if if *nvmname* of the addressed optical module equals OM10 and the port attribute *istd* equals SDH. This are reported by means of DLT-STM64 notification of port.

11. One instances of port OC-192 is deleted if *nvmname* of the addressed optical module equals OM10 and the port attribute *istd* equals SONET. This are reported by means of DLT-OC192 notification of port.
12. The parameters *nvmname*, *nvmqual* of the slot will be set to UNDEFINED, UNDEF, which will be reported by an ED-EQPT notification for slot.

Update affected entities - GE10PL1

The command shall update / delete in addition a set of entities of different types depending on the provisioned optical module:

1. All related tributaries are deleted for the VCG port related to the optical module. This are reported by means of DLT-VCGTRIB notifications.
2. The VCG related to the optical module is deleted. This are reported by means of a DLT-VCG notification.
3. The Ethernet port related to the optical module is deleted. This are reported by means of a DLT-EPORT notification.
4. The parameters nvmname, nvmqual of the socket will be set to UNDEFINED, UNDEF, which will be reported by an ED-EQPT notification for slot.
5. If the last optical module is de-provisioned from the GE10PL1 pack, then the acceptable module lists of all sockets on the pack shll be updated to the full range of allowed optical modules.

Group notification for STS-1/AU-3 tributaries

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to 1&&-<N>, with N is maximal legal STS-1/AU-3 number.

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

Disabling of PM monitor

See Detailed behavior specification in "ED-STM16: Edit STM16 port" (p. 3-1035) w.r.t PM impact as side of effect of port/tributary deletion.

Example input/output

This example shows a DLT-EQPT command de provisioning a socket:

```
DLT-EQPT:UNITE-NE:1-1-#-#-5-sc2:123456;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-EQPT** command.

Validation rule: No cross-connections allowed

Check of cross-connections involving the module to be removed.

Error code:

`SNVS – Status, Not in Valid State, cross-connections are still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure allowed

Check if lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure exist which involve the module to be removed. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, but no higher order cross-connections are present, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: No XC loopbacks allowed

Check of XC loopbacks involving the module to be removed.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: No facility loopbacks allowed

Check of facility loopbacks involving the module to be removed.

Error code:

`SNVS – Status, Not in Valid State, facility loopbacks still exist`

Validation rule: No protection groups allowed

Check of protection groups involving the module to be removed.

Error code:

`SNVS – Status, Not in Valid State, protection group still exist`

Validation rule: No transparent DCC connections

Check of used transparent DCC connections involving the module to be removed.

Error code:

`SNVS – Status, Not in Valid State, transparent DCC connections are still existing`

Validation rule: Not used as timing reference

Check of that none of the ports of the involving module is used as timing reference.

Error code:

`SNVS – Status, Not in Valid State, at least one port used as timing reference`

Validation rule: Not used for OW

Check of that no connection to a OW port exists.

Error code:

`SNVS – Status, Not in Valid State, connection to OW port exists`

Validation rule: module must be plugged out

Check of that the no module is plugged into the involved socket.

Error code:

`SNVS – Status, Not in Valid State, the Module is not plugged out`

Validation rule: Check whether system is in maintenance condition

Deprovisioning is not allowed, when the system in is maintenance condition.

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Check if related ports are not in use by ONNS

If the provisioned module has ports with *portClass* = INNI or *portClass* = EDGE with *networkTNAType* = IPV4 and *networkTNAAddress*=0.0.0.0 the following must be fulfilled:

1. No cross-connection may exist on any tributary of the port.
2. *portControl* must be LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Validation rule: Check if related ports are not in use by ONNS

De-provisioning is not allowed if the provisioned module has ports with *portClass* = UNI or *portClass* = eNNI or *portClass* = EDGE with *networkTNAAddress*≠default value.

Error code:

`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Validation rule: No test access session

De-provisioning is not allowed if one or more tributaries on the port related to the socket are used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: Socket must be provisioned

When de-provisioning a module it is checked whether the socket is provisioned (i.e. *nvmname*/*nvmqual* is different from UNDEFINED/UNDEF).

Error code: `SNVS – Status, Not in Valid State, socket is not provisioned`

Validation rule: Equipage status and inventory data of module must be accessible

If the equipage status or inventory data of a module is not accessible (e.g. due to the parent pack being in recovery) de-provisioning of the socket is not allowed.

Error code: `SNVS – Status, Not in Valid State, unknown module equipage`

Related TL1 messages

- "RTRV-EQPT (socket): Retrieve equipment (socket)" (p. 3-2391)
- "ED-EQPT (socket): Edit equipment" (p. 3-379)
- "ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
- "RTRV-STATE-EQPT (socket): Retrieve equipment state " (p. 3-2842)

DLT-EQPT (CP) notification

Output format

Syntax **DLT-EQPT**:*aid*::

Output parameters

Command parameters

1. **aid** – Access Identifier
   Description: AID of the deleted circuit pack.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for circuit pack in any slot
   - LoxcAID: ::= UnivSlotAID-loxc for LOXC pack in universal slots

   The keyword **usall** is not allowed for Simple_UnivSlotAID. The keyword **xcall** is not allowed for Simple_XCSlotAID. The keyword **ctlall** is not allowed for Simple_CTLSlotAID. The keyword **all** is not allowed for Simple_SlotAID.

Example output

```
DLT-EQPT:UNITE-NE:1-1-#-#-1:123456;

   UNITE-NE 00-11-20 08:00:00
M  123456 COMPLD
;

   UNITE-NE  00-11-20  8:01:03
A  610  REPT  DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-EQPT:1-1-#-#-1-cp:::″
;

   UNITE-NE  00-11-20  8:01:03
A  611  REPT  DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-OC768:1-1-#-#-1-1:::″
;

   UNITE-NE  00-11-20  8:01:03
A  612  REPT  DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-VC3:1-1-#-#-1-1-1&&-768:::″
;

   UNITE-NE  00-11-20  8:01:03
A  613  REPT  DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-FECOM:1-1-#-#-1,LINE:::″
;

UNITE-NE  00-11-20  8:01:03
```

```
A  613  REPT  DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-FECOM:1-1-#-#-1,SECTION:::”
;

   UNITE-NE  00-11-20  8:01:03
A  613  REPT  DBCHG
   ”Time=8-00-30,DATE=00-11-20:ED-EQPT:1-1-#-#-1::nvmname=UNDEFINED,nvmqual=UNDEF:”
;
```

DLT-EQPT (socket) notification

Output format

Command syntax: **`DLT-EQPT`***:aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keywords all / usall / scall shall are not allowed on any level.

Example output

This example shows a DLT-EQPT notification reporting the deletion of a socket:

```
   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-14:DLT-EQPT:1-1-#-#-5-sc2”
;
```

DLT-FECOM notification

Output format

The **DLT-FECOM** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **DLT-FECOM** notification has the following specific syntax:

DLT-FECOM::*port*,*dcctype*:*spec_block*

Output parameters

Command parameters

1. **port**
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID
   Range: SPortAID. The keyword 'all' is not allowed.
2. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC of the port is affected by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.
   Default:
   - SECTION: if dccstat is set to ENABLE
   - no default (i.e. parameter is mandatory) if scn_dccstat or uni_dccstat is ENABLE.

Spec block parameters

1. **osinode**
   Assignment of a DCC channel to an MCN node in the NE. The parameter is omitted when dccstat != ENABLE.
   An UNITE NE can have up to 8 instances of the OSI stack. Node-1 is default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00). Nodes 2-8 can be created and configured, see command **ENT-ULSDCC-L3**.
   Type: AID
   Range: osinode1-8.

Example output

```
   NE-UNITE 01-01-08 01:10:11
A  000 REPT DBCHG
   ″TIME=01-01-09,DATE=00-00-13:DLT-FECOM::1-1-#-#-5-1,LINE:osinode1″
;
```

DLT-IP-MAP: Delete TCP/IP map

The **DLT-IP-MAP** command can be used to delete one entry in the TCP/IP application context ID map which was created via ENT-IP-MAP. The tl1tcpipgw parameter in the **ED-NE** command enables or disables the mapping feature.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-IP-MAP**:*tid::ctag:::spec_block;*

Input Parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: TID.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: CTAG.

Spec block Pparameters

1. **ip** – IP address
   This parameter is optional. If it is omitted the mapping is not changed and a normal COMPLD message is returned.

Output format

If the DLT-IP-MAP request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command shall delete the mapping entry if the ip parameter was specified and could be found in the TCP/IP application context ID mapping.

Example input/output

The following example shows the result of an DLT-IP-MAP command:

```
DLT-IP-MAP:LT-UNITE::123456:::ip=198.78.46.8;

   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`DLT-IP-MAP`** command.

Validation rule: Check for known IP address

The removal of an entry from the IP map is only allowed if entries with the specified IP address are used within the IP map.

Error code:

`SNVS – Status, Not in Valid State, unknown IP address within MAP`

Related TL1 Messages

- “RTRV-IP-MAP: Retrieve TCP/IP map” (p. 3-2436)
- “ENT-IP-MAP: Enter TCP/IP map” (p. 3-1594).

DLT-IP-MAP notification

Output format

The **`DLT-IP-MAP`** command shall not generate a REPT DBCHG notification. This is due to a potential security hazard: any listening manager who receives this notification could otherwise fake its own IP address to get access to the T-TD.

DLT-LONIM notification

Output format

Command syntax:

ED-LONIM :*aid*;

Output parameters

Command parameters

1. **aid** – Access Identifier
   The access identifier specifies a low-order tributary to be operated on.
   Type: AID
   Range: GenLOTribAID, DS1NumAID.
   The following values are not used:
   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID
   - *all* in Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
   - *all* in Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID

   Default: None.

Example output

Example of a DLT-LONIM notification removing a SDH subtructure:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-LONIM:1-1-#-#-39-16-1-1&&-63"
;
```

Example of a DLT-LONIM notification removing a SONET subtructure:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-LONIM:1-1-#-#-39-16-3-1&&-28"
;
```

DLT-LOPTF notification

Output format

Command syntax: **`DLT-LOPTF`**:*aid*:;

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: TMuxSLONumAID.
   The following values are not used:
   - **`all`**, **`usall`** for Simple_UnivSlotAID
   - **`all`** for Simple_UnivSlotAID
   - **`all`** for Simple_TMuxSPTNumAID
   - **`all`** for Simple_TMuxSLONumAID

   No wildcards are allowed.
   Default: None.

Example input/output

Example of a **`DLT-LOPTF`** notification removing a SONET substructure:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-LOPTF:1-1-#-#-39-tms16-1-1&&-28"
;
```

DLT-MANROUTE-LAN: Delete manual route at LAN port

The **DLT-MANROUTE_LAN** command is used to delete manual routes at LAN ports, which are used as UNI control channels.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-MANROUTE-LAN**:*tid*:*aid*:*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*.

Spec block parameters

1. **ip_ad** – destination IP Address
   This parameter is used to indicate the destination IP address to which a manual route is created. This is the LAN i/f address of UNI-C
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – Node ID of UNI-C
   This parameter is used to indicate the router ID of the UNI-C to which a manual route is created.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255. The values 0.0.0.0 and 1.1.1.1 are not allowed.
3. **osinode**
   This parameter is used toindicate the OSI node assigned to the LAN port.
   Type: AID.
   Range: uninode.

Output format

If the **DLT-MANROUTE-LAN** request is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **DLT-MANROUTE-LAN** command deletes a manual route entry for a LAN port, which is used as control channel for UNI.

The deletion of the manual route is notified with a **DLT-MANROUTE-LAN** notification.

Example input/output

The following example shows the result of **DLT-MANROUTE-LAN**:

```
   NE-UNITE 06-06-30 09:00:01
A  000 REPT DBCHG
   ″TIME=09-00-01,DATE=06-06-30:DLT-MANROUTE-LAN:1-1-lan2::ip_ad=135.248.206.64, ip_
      ad_uni=135.248.206.64″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`DLT-MANROUTE-LAN`** command.

Validation rule: Not allowed to use LAN1 for UNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for UNI`

Validation rule: LAN port not used for UNI

If the LAN port specified in the AID is not in use for UNI, i.e. is not assigned to the uninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, LAN port not in use for UNI`

Validation rule: Manual Route entry must exist

If the manual route entry specified by *ip_ad* and *ip_ad_uni* does not exist for the LAN port, then the command is denied.

Error code:
`IDNV – Input, Data Not Valid, Manual route entry does not exist`

Related TL1 messages

- "RTRV-MANROUTE-LAN: Retrieve manual route at LAN port" (p. 3-2516)
- "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610)

DLT-MANROUTE-LAN notification

Output format

Command syntax: **`DLT-MANROUTE-LAN`**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.

Spec block parameters

1. **`ip_ad`** – destination IP Address
   This parameter is used to indicate the destination IP address to which a manual route is created. This is the LAN i/f address of UNI-C
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **`ip_ad_uni`** – Node ID of UNI-C
   This parameter is used to indicate the router ID of the UNI-C to which a manual route is created.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.

Example input/output

The following example shows the result of DLT-MANROUTE-LAN:

```
   NE-UNITE 06-06-30 09:00:01
A  000 REPT DBCHG
   "TIME=09-00-01,DATE=06-06-30:DLT-MANROUTE-LAN:1-1-lan2::ip_ad=135.248.206.64, ip_
      ad_uni=135.248.206.64"
;
```

DLT-MANROUTE-MCN: Delete one manual MCN IP route entry

The **DLT-MANROUTE_MCN** command is used to delete a manual MCN IP route entry.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: Yes.

Input format

Command syntax: **DLT-MANROUTE-MCN**:*tid*:*aid*:*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   The keyword "lanall" is allowed for Simple_LANPortAID.
   Default: Omitting this field has the same effect as the keyword "lanall", that means, the report will display the manual MCN route entries of all possible LAN ports assigned to the MCN.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **dest_ipad** – Destination IP Address
   The IP address of an IP node which can be reached via one of the LAN ports.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: None.

Output format

If the **DLT-MANROUTE-LAN** request is successful, the following normal completion response is returned:

```
sid date time
```

```
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **DLT-MANROUTE-MCN** command deletes a manual MCN IP route entry from the system.

The deletion of the manual route is notified with a **DLT-MANROUTE-MCN** notification.

Example input/output

The following example shows the result of **DLT-MANROUTE-MCN**:

```
DLT-MANROUTE-MCN:UNITE-NE:1-1-lan2:123456:::dest_ipad=192.168.90.4;

   UNITE-NE 08-06-27 17:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **DLT-MANROUTE-MCN** command.

Validation rule: Manual MCN IP Route entry must exist

The command is denied if a manual MCN IP route entry which has the same value in the parameter *dest_ipad* does not exist for the specified LAN port.

Error code:

`IDNV – Input, Data Not Valid, Manual route entry does not exist`

Related TL1 messages

- “RTRV-MANROUTE-MCN: Retrieve manual IP route for management communication” (p. 3-2519)
- “ENT-MANROUTE-MCN: Enter manual IP route for management communication” (p. 3-1614)
- “ENT-MANROUTE-MCN notification” (p. 3-1618)
- “DLT-MANROUTE-MCN notification” (p. 3-209)

DLT-MANROUTE-MCN notification

Output format

Command syntax: **DLT-MANROUTE-MCN**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   The keyword "lanall" is allowed for Simple_LANPortAID.
   Default: Omitting this field has the same effect as the keyword "lanall", that means, the report will display the manual MCN route entries of all possible LAN ports assigned to the MCN.

Spec block parameters

1. **dest_ipad** – Destination IP Address
   The IP address of an IP node which can be reached via one of the LAN ports.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: None.
2. **dest_ipmask** – Destination IP Subnet Mask
   The IP subnet mask of the IP node which can be reached via the LAN port.
   Type: Four dot separated numeric strings
   Range: Each numeric string: 0 … 255.
   Default: None.
3. **nexthop_ipad** – Next Hop IP Address
   The IP address of the next hop routing node via which the Destination IP Address can be reached.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The value 0.0.0.0 is used if the IP node defined by *dest_ipad* is connected to the LAN port directly. The values 1.1.1.1 and 127.0.0.1 are not allowed.
   Default: None.

Example input/output

The following example shows the result of **DLT-MANROUTE-MCN**:

```
   UNITE-NE 08-06-27 17:00:02
A  000 REPT DBCHG
   "TIME=17-00-00,DATE=08-06-27:DLT-MANROUTE-MCN:1-1-lan2::dest_
      ipad=192.168.90.4,nexthop_ipad=192.168.2.4,nexthop_ipmask=255.255.255.0"
```

;

DLT-OC3 notification

Output format

The **DLT-OC3** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **DLT-OC3**:*aid*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: OC3 port AID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-OC3:1-1-#-#-4-1
;
```

DLT-OC12 notification

Output format

The **DLT-OC12** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **DLT-OC12**:*aid*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: OC12 port AID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-OC12:1-1-#-#-4-1
;
```

DLT-OC48 notification

Output format

The **DLT-OC48** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **DLT-OC48**:*aid*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: OC-48 port AID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-OC48:1-1-#-#-4-1
;
```

DLT-OC48T notification

Output format

Syntax

`DLT-OC48T`*:aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: OC-48 port AID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-OC48T:1-1-#-#-4-1
;
```

DLT-OC192 notification

Output format

Syntax: **`DLT-OC192`**:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: SPortAID of OC192 port. The keyword **`all`** is not allowed.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-OC192:1-1-#-#-4-1”
;
```

DLT-OTU2 notification

Output format

Command syntax: **`DLT-OTU2`**:*aid*

Output parametes

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OTU2 port.
   Type: AID
   Range: OTNPortAID of OTU2 port.
   The keyword ″otall″ is not allowed.

Example output

```
   UNITE-NE 06-06-30 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=06-06-30:DLT-OTU2:1-1-#-#-4-1″
;
```

DLT-OUC: Delete ONNS to UPSR/SNCP constructs

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax:

DLT-OUC:*tid:wrk_aid:ctag[::ouc_dir];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **wrk_aid** – Working Tributary Access Identifier
   This parameter specifies the working tributary of the OUC.
   Type: STS1NumAID
   Range: n.a. (**all** values shall never occur as part of an output parameter)
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **ouc_dir** – OUC Directionality
   This parameter specifies the directionality of the OUC to be deleted via the **DLT-OUC** command. This parameter will be needed especially, if one of two existing unidirectional OUCs are deleted.
   Type: Alphanumeric value set.
   Range:
   - BIDIR (default)
   - UNIDIR-ADD
   - UNIDIR-DROP.

Output format

If the **DLT-OUC** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Example input/output

A bidirectional OUC between working tributary 4 of a 10G SDH EDGE port in slot 01 and protection tributary 7 of a 2G5 SDH EDGE port 3 in slot 34 will be deleted by the following command:

```
DLT-OUC:UNITE-NE:1-1-#-#-1-1-4:123456::BIDIR;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

A unidirectional drop OUC between working tributary 4 of 2G5 SONET EDGE port 1 in slot 10 and protection tributary 7 of a 2G5 SONET EDGE port 3 in slot 34 will be deleted by the following command:

```
DLT-OUC:UNITE-NE:1-1-#-#-10-1-4:123456::UNIDIR-DROP;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Entity (OUC) must exist

OUC command processing shall verify that the specified OUC exists.

Error code: `IENE – Input, Entity Not Exists, Specified OUC does not exist`

Validation rule: No forced XC loopback must exist on EDGE-wrk or EDGE-prt tributary

OUC command processing shall deny requests to delete an OUC if, on the EDGE-wrk or EDGE-prt tributary, a forced cross-connection loopback exists.

Error code:

`SNVS – Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary`

DLT-OUC notification

The DLT-OUC notification will notify only the deletion of the OUC itself. cross-connections and path protection groups that are deleted implicitly via an OUC deletion is notified via corresponding DLT-CRS-modifier and DLT-PROTN-GRP notifications.

Output format

Syntax:

`DLT-OUC`:*wrk_aid,prt_aid:ouc_dir;*

Output parameters

Command parameters

1. **`wrk_aid`** – Working Tributary Access Identifier
   This parameter specifies the working tributary of the OUC.
   Type: STS1NumAID
   Range: n.a. (**`all`** values shall never occur as part of an output parameter)
2. **`prt_aid`** – Protection Tributary Access Identifier
   Protection Tributary Access Identifier. Specifies the protection tributary of the OUC.
   Type: STSNumAID
   Range: n.a. (**`all`** values shall never occur as part of an output parameter)
3. **`ouc_dir`** – OUC Directionality
   Directionality of the OUC
   Type: Alphanumeric value set.
   Range:
   - BIDIR (default)
   - UNIDIR-ADD
   - UNIDIR-DROP.

Example input/output

```
   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:DLT-OUC:1-1-#-#-1-1-1,
   1-1-#-#-2-1-1:BIDIR″
;


   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:DLT-OUC:1-1-#-#-1-1-1,
   1-1-#-#-2-1-1:UNIDIR-ADD″
;
```

```
   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:DLT-OUC:1-1-#-#-1-1-1,\
   1-1-#-#-2-1-1:UNIDIR-DROP″
;
```

DLT-PATH: Delete ONNS path connection

Overview

The **DLT-PATH** command can be used to delete all members of a connection group with one single command.

An ONNS connection may include multiple paths as in the case of a network 1+1 connection that contains an active path and a standby path, each with its own *pathID*. A path connects 2 client Edge ports (or *connPoints*) through a series of NE nodes from a source node to a destination node to provide a communication service.

The **DLT-PATH** command deletes all paths associated with a *connID* or alternatively specified by its *connAlias*. The ONNS connection must begin at the node to which the message was addressed. The NEs along the path controlled by ONNS shall remove the local cross-connection and return resources to the available pool.

The **DLT-PATH** command deletes the worker path associated with a *connID* or alternatively specified by its *connAlias*. The ONNS connection must begin at the node to which the message was addressed. The Network Elements along the path controlled by ONNS shall remove the local cross-connection and return resources to the available pool. A protection path as part of the associated protection connection or of the worker connection of the deleted working connection might be also removed depending on the needed protection paths of the current M:N protected connection group.

The **DLT-PATH** command for connection groups generates one separate **REPT-PTHCHG** message for each connection successfully deleted.

Successful connection deletion with forced cross-connection loopbacks existing

If the network element receives a **DLT-PATH** command for a connection and on top of this connection a forced cross-connection loopback exists in the A- or Z-node, deletion of the connection shall be performed as usual with keeping the existing forced cross-connection loopback up as normal loopback on the corresponding tributary.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No

Input format

Command Syntax of DLT-PATH to delete a path from a connection:
DLT-PATH:*srcTid:[srcAid]:ctag::[srvID]:[connID][:specBlock];*

A user defined connection alias (an alternate name) may be optionally specified with the **DEL-PATH** command. The alias is specified in the spec block as *connAlias*.

If no *connAlias* is specified, the *connID* is used as the key. If both *connAlias* and *connID* are specified then *connID* is used as the key. If neither of the two is specified the command is denied.

The *connAlias* is returned as an output parameter in the response along with all the other parameters. If it not defined, a NULL is returned.

Deletion of a whole connection group is not possible using a connection alias as input.

Deletion of single worker connection being member of a M:N protected connection group is allowed by specifying the *connID* or the *connAlias* as the key. Deletion of a protection connection is denied.

Deletion of single connections being member of a switched VCG connection group (VCG connection group initiated by UNI) is rejected.

Deletion of a whole GbE group of connections shall be rejected, if the affected VCG has port class EDGE and at least one of the affected connections has connControl value EXT.

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.

4. **`srvID`** – Service Identifier
   This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.

5. **`connID`** – Connection ID
   Connection ID for an existing ONNS path connection service. Reports only the connections originating at the specified source node. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.
   Type: Character string of length up to 39.
   Range: *ALL* **–** special value to retrieve all connection records

   Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1..39), Port # (1..16), Tributary # (1..192)
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x. Example: 29.13.196.203&1-1-13-15-1&271584.

   Range: *nodeID&Tributary AIDrange* - if port # specifies a VCG to retrieve a connection group, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1..39), Port # (1..9), Tributary # ALL- special value to retrieve all connection records associated with the port ID.

   As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG. The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-5-13-7-ALL.
   The port AID 1-5-13-7 must denote a VCG.

   Range: *nodeID&default Tributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - defaultTributary AID: five dash separated numeric strings with: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##
   - Index: Numeric string 0-4294967295.

   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&proupIndex* - to retrieve all connections belonging to an M:N protected connection group, with:

- nodeID (four dot separated numeric strings: each string 0..255)
- groupIndex (integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Spec block parameters

1. **`connAlias`**
   This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83.
   Range The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.

Output format

Normal system response to DLT-PATH for an NN connection of any of the supported protection types

If the **`DLT-PATH`** request completes successfully, the normal completion response is returned as shown below:

```
DLT-PATH:srcTid:srcAid:ctag::[srvID]:connID;

   srcTid date time
M  ctag COMPLD
   ″ConnID=connID,ConnStatus=DLT,,,,
   [ConnAlias=connAlias,]
   ,[SrvID=srvID],,,,,,,
   ,,,″
;
```

System response format to DLT-PATH when deleting groups of connections

For the **DLT-PATH** request for deleting a group of connections the response is as follows: down response for connection group:

- If the command as a whole is denied due to e.g. syntax or input failure, the command is denied simply giving completion code DENY together with the error reason
- If deletion of all connections of the group succeeded, the command is completed giving completion code COMPLD and all connections of the group is listed. For each connection the parameters are given as would be given in a reply to a successful **DLT-PATH** command for a single connection.
- If not all connections could be deleted successfully, but the deletion of a number of connections failed, the command is completed giving completion code PRTL and giving a list of the following informations: For each connection whose deletion succeeded the parameters are given as would be given in a reply to a successful **DLT-PATH** command for a single connection. For each connection whose deletion failed the connection ID is given together with the error reason for the failure as would be given in a reply to a failed **DLT-PATH** command for a single connection. This shall also hold, if the deletion of all connections failed individually.

System response format to DLT-PATH when deleting a single worker connection of a M:N protection group

For the **DLT-PATH** request for deleting a single worker connection of a M:N protection group the response is as follows:

- If the command as a whole is denied due to e.g. syntax or input failure, the command is denied simply giving completion code DENY together with the error reason
- If deletion of the group worker connection and a possible protection connection succeeded, the command is completed giving completion code COMPLD and both connections are listed. For both connections the parameters are given as would be given in a reply to a successful **DLT-PATH** command for a single connection.
- If the worker connection could be deleted successfully, but a possible deletion of the protection connection failed or vice versa, the command is completed giving completion code PRTL and giving a list of the following informations: For he connection whose deletion succeeded the parameters are given as would be given in a reply to a successful **DLT-PATH** command for a single connection. For the connection whose deletion failed the connection ID is given together with the error reason for the failure as would be given in a reply to a failed **DLT-PATH** command for a single connection.
  This shall also hold, if the deletion of both connections failed individually.

In case of an M:N protection connection group a partial completion of the deletion of the group in detail means the following:

- The protection group does not work any more
- For connections that could not be deleted M:N protection group related parameters may still indicate valid values for group operation
- For such connections the parameter connStatus will indicate that the group is not working any more
- As for all types of connections deletion will be retried regularly until deletion finally succeeds.

The **`DLT-PATH`** command for grouped connection deletion shall send a maximum of 8 connection records per block. On all large output message blocks but the last, the continuation completion code COMPLD, PRTL or DENY is used depending on the outcome for the whole command and the semicolon(;) is replaced with a (>) terminator. Then on the last record block, the COMPLD completion code COMPLD, PRTL or DENY is used depending on the outcome for the whole command with the final semicolon (;) data block terminator.

Note: This **`DLT-PATH`** command for grouped connection deletion can produce large amounts of data. This will require proper sizing of the receive buffer of the OS system.

Output parameters

Command parameters

1. **`srcTid`**
   See the input parameter description above.
2. **`date`**
   This parameter is set to the date when the output message is generate: YY-MM-DD where YY=00-99, MM=01-12, DD=01-31.
3. **`time`**
   This parameter is set to the time when the output message is generated: HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.
4. **`ctag`** – Correlation Tag
   See the parameter description in the Input parameters section above.
5. **`completionCode`**
   Standard TL1 completion code result.
   Type: Value set.
   Range:
   - COMPLD: for success (default)
   - PRTL: for partial success / failure, see section ERROR RESPONSES for details
   - DENY: in case of errors, see section ERROR RESPONSES for details.

6. **`connID`**
   See the parameter description in the Input parameters section above.
7. **`connAlias`**
   See the parameter description in the Input parameters section above.
8. **`connStatus`**
   ConnStatus = pst&sst(&tst) Connection status is the overall status of the connection connID which may be different than the status of an individual *pathID* in that connection.

   connStatus is composed out of

   - pst (PrimaryState)
   - sst (SecondaryState)
   - tst (TertiaryState, optional).

   Type: up to three &-separated value sets.

   Range *pst* values:

   - CON: Connected
   - DLT: Deleted
   - NOCON: Not Connected
   - MAINT: Maintenence Mode
   - UNKNOWN: Conn state unknown.

   Range *sst* & *tst* values

   - OK: Conn good
   - FAILED: Conn failed
   - SETUP: Conn SetUp occurring
   - TRDOWN: Conn teardown occurring
   - NO-IN: Conn failed, signal in bad
   - RSTOR: path restore occurring
   - RSTRD: Conn restored
   - RSTRFL: Conn restore failed
   - DLT-FL: Delete operation Failed
   - UNKNOWN: Conn state unknown
   - WTORSTR: Waiting to restore path
   - PREMPT: Path pre-empted
   - SETUPFL: Connection SetUp Failed
   - FL-NVM: Not saved in NVM
   - WTORV: Waiting to revert path
   - EPRTFL: EDGE Port Failed

- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection).
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection).

Range *tst* values

- DEGRADED: Non-revertive restoration was not possible if the worker or protection path of a permanent 1+1 protected connection fails

9. **`srvID`** – Service Identifier
   See the input parameter description above.

Example input/output

Input/output example of deleting a single NN connection of any protection type

The following example shows the successful completion of a single NN connection deletion by a **`DLT-PATH`** command:

```
DLT-PATH:LT_ONNS_789:snn:433456::135.7.9.2-46:192.1.165.12&1-6-5-3-1&1;

   LT-ONNS-789 2002-07-01 08:00:00
M  433456 COMPLD
   ″ConnID=192.1.165.12&1-6-5-3-1&1,ConnStatus=DLT,,,,
   ConnAlias=AEXP-NY-BOS-ckt,
   ,,,,,,,,
   ,,,″
;
```

Input/output example of deleting a single M:N worker connection

The following example shows the successful completion of a single M:N worker connection deletion with an implicit M:N protection connection deletion by a **`DLT-PATH`** command:

```
DLT-PATH:LT_ONNS_789:snn:433456::135.7.9.2-46:135.13.196.203&1-1-13-4-1&1;


   LT-NN-789 2002-09-18 08:00:00
M  973456 COMPLD

   ″ConnID=135.13.196.203&1-1-13-4-1&1,ConnStatus=DLT,,,,
   ConnAlias=AmericanExp-NY-BOS-ckt_wrk_001,
   ,,,,,,,,
```

```
   ,,,"


   "ConnID=135.13.196.203&1-1-#-#-#&3,ConnStatus=DLT,,,,
   ConnAlias=AmericanExp-NY-BOS-ckt_prt_001,
   ,,,,,,,,
   ,,,"
;
```

Connection deletion, deletion in the network failed

If the NE receives a **DLT-PATH** command for a connection and this connection specified is not deleted in the network due to signaling problems, ONNS will not remove this connection from the ONNS data repository, but nevertheless return the following positive reply. In case of multiple connections affected by the **DLT-PATH** command for each connection that failed to be deleted a corresponding entry is contained in the response besides the entries for the connections that succeeded.

```
   srcTid 2001-07-01 08:00:00
M  ctag CMPLD

   "ConnID=135.13.196.203&1-1-13-4-1&1,ConnStatus=DLT-FL,,,,
   ConnAlias=AmericanExp-NY-BOS-ckt,
   ,,,,,,,,
   ,,,"
;
```

Additionally for each connection, the deletion of which failed, a **PTHCHG** notification for the state change to **DLT-FL** will be sent. The **DLT-PATH** command is accepted in such cases, because an affected ONNS connection will be automatically deleted when signaling is possible again. No further activity by the operator is necessary.

Error responses

Validation rule: Path *ConnectionID* does not exist

If the Source Node ONNS receives a DLT-PATH command when the specified connection path does not exist, then the following error response is returned:

Error code: `E220 – 135.13.7.9-48 – Input,ConnectionID Non-Existent`

The following list shows conditions that will cause similar error responsse:

Error code: `E037 – Input, Neither ConnID nor ConnAlias Specified`

Error code: `E221 – IANE Error – Input, connAlias Non Existant`

Error code: `E038 – IICS – Input, Inconsistent connAlias Specified`

Validation rule: Connection is member of an M:N protected connection group

If the Source Node ONNS receives a DLT-PATH command with the specified connection ID denoting a working or protecting connection of an M:N protected connection group, then the following error response is returned:

Error code:

`E222 – IIDM srvID - Input, Invalid deletion of connection of protection`

Validation rule: Protection connection of an M:N protection group in use

If the source node ONNS receives a DLT-PATH command with the specified connection ID denoting a working connection of a M:N protection group which also would lead to a implicit tear down of a protection connection currently used by another working connection than the addressed one is denied:

Error code:

`E223 – SPCU 135.7.9.2-35 – Status, Protection Connection in Use, removal of the M:N worker connection aborted`

Validation rule: Connection is member of a switched VCG connection group

If the Source Node ONNS receives a DLT-PATH command with the specified connection ID denoting a connection of a switched VCG connection group (VCG connection group initiated by UNI), then the following error response is returned:

Error code:

`E224 – IIDC srvID – Input, Invalid Deletion of a Connection of a switched connection group`

The following list identifies conditions that will cause error responses specific to DLT-PATH commands to delete groups of connections:

Error code:

`E050 – IICG – Input, Invalid connection ID for connection group/connID`

Validation rule: Grouped connection deletion on VCG EDGE affects EXT Connection

If the Source Node ONNS receives a DLT-PATH command requesting a GbE group deletion on an EDGE VCG affecting an ONNS connection with EXT connControl, then the following error response is returned:

??? Error code:

`E225 - IIGD srvID - Input, Invalid Grouped Deletion of EXT Connection`

The following list identifies conditions that will cause error responses specific to DLT-PATH commands to delete groups of connections:

Error code:

`E050 – IICG – Input, Invalid connection ID for connection group/connID`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "ED-NNCNFG: Modify ONNS system configuration" (p. 3-450)

DLT-PROTN-GRP: Delete protection group

The **DLT-PROTN-GRP** command allows to delete protection groups which were previously created via **ENT-PROTN-GRP**.

Some types of protection groups are deleted in the context with another stimuli. **DLT-PROTN-GRP** cannot be used to delete the following types of protection groups:

- switching and controller equipment protection group (deleted as part of delete circuit pack)
- path protection group (deleted as part of **DLT-CRS** for path-protected XC).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

Equipment protection group

The command **DLT-PROTN-GRP** is applicable for the EP155, EP51, LOXC, TransMUX, and ES64U equipment protection.

Command syntax: **DLT-PROTN-GRP**:*tid*:*aid*:*ctag*::*protype*;

Multiplex section/line protection group

Command syntax: **DLT-PROTN-GRP**:*tid*:*aid*:*ctag*::*protype*;

2-/4-fibre MS-SPRing/BLSR protection group

Command syntax: **DLT-PROTN-GRP**:*tid*:*[aid]*:*ctag*::*protype*, *[rid];*

Input parameters

Command parameters (1+1 equipment protection group)

1. **tid** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of the equipment protection group.
   Type: AID
   Range: EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID.
3. **ctag** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).

4. **`protype`** – Protection Type
   The parameter evaluation is case sensitive.
   This parameter specifies the type of the protection group.
   Type: Alphanumeric value set.
   Range:
   - EQPTSWFBR (if *aid* is SwitchAndTmgProtGrpAID)
   - EQPTCTL (if *aid* is CtlProtGroupAID)
   - EQPTEP155 (if *aid* is EP155ProtGroupAID)
   - EQPTEP51 (if *aid* is EP51ProtGroupAID).
   - EQPTLOXC (if *aid* is LOXCProtGroupAID)
   - EQPTTMUX (if *aid* is TMUXProtGroupAID)
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

Command parameters (multiplex section/line protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port aid being a member of the group.
   Type: AID
   Range: OnePlusOne_ProtGrpAID (not ranged) or NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **`ctag`** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **`protype`** – Protection Type
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later).
   Type: value set.
   Range: 1+1 or 1xNOPT.

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged).
3. **`ctag`** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F.
5. **`rid`** – Ring Identification
   The parameter specifies the ring identification name. Storage and retrieval of the rid is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Option: If *aid* is specified, then *rid* can be omitted.

Output format

After receiving the **`DLT-PROTN-GRP`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

Refer to the “Output parameters” section of the “RTRV-HDR: Retrieve header” (p. 3-2417) command for a normal completion response.

Detailed behavior description

LOXC 1+1 equipment protection group

The command removes/deletes the specified LOXC protection group if protype equals EQPTLOXC. The deletion of the protection group is notified by means of a DLT-PROTN-GRP notification. In addition, the low order switch capacity of the protection unit is set back to 0 GBit/s and the LO tributary standard is set to UNDEFINED, which causes a referring ED-EQPT notification.

The existence of transmission entities is not influenced by the deletion of the LOXC EQP group since they will be only controlled by low-order cross-connection commands.

Precondition for that is that the worker LOXC pack is active. If the protection LOXC pack is active, then the LOXC protection group can only be deleted if no transmission entity is still present.

TransMUX 1+1 equipment protection group

The command removes/deletes the specified TransMUX protection group if *protype* equals EQPTTMUX. The deletion of the protection group is notified by means of a **`DLT-PROTN-GRP`** notification. In addition, the TransMUX switch capacity of the protection unit is set back to 0 GBit/s. This are notified via **`ED-EQPT`** notification for the circuit pack.

Precondition for that is that the worker TransMUX pack is active. If the protection TransMUX pack is active, then the TransMUX protection group can only be deleted if no HO or LO cross-connection is still present.

ES64U 1+1 equipment protection group

The command removes/deletes the specified ES64U protection group if *protype* equals EQPTES64U. The deletion of the protection group is notified by means of a **`DLT-PROTN-GRP`** notification. In addition, the ES64U switch capacity of the protection unit is set back to 0 GBit/s. This are notified via **`ED-EQPT`** notification for the circuit pack.

Note: The IP address of ES64U protection pack *dcn_ipad* is not changed when deleting the group.

Precondition for that is that the worker ES64U pack is active. If the protection ES64U pack is active, then the ES64U protection group can only be deleted if no HO cross-connection is still present.

EP155 1+1 equipment protection group

The command removes/deletes the protection group of type as specified by means of parameter protype. If *protype* equals ’EQPTEP155’, then one instance of EP155 equipment protection group is deleted. The deletion of the protection group is reported by means of an **`DLT-PROTN-GRP`** notification.

EP155 equipment protection group - port impact due to EP155 protection group deletion

Ports related to the protection card of the protection group is recreated with default values, because they had been deleted at the enter time of the protection group. This creation is reported by means of a set of **ENT-STM1E** notifications.

EP155 equipment protection group - tributary creation due to EP155 protection group deletion

ributaries related to the protection card of the protection group is recreated with default values, because they had been deleted at the enter time of the protection group. This creation is reported by means of a set of ENT-VC3 notifications if the protection port is an SDH port.

The number of tributaries to be created is derived from the port rate. For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to 1&&-<N> with N is maximal legal STS-1/AU-3 number.

EP155 equipment protection group - AID range for group notification

See STS-1/AU-3 Tributaries AID w.r.t. range for legal STS-1/AU-3 number.

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192

EP51 equipment protection group - 1+1 Equipment Protection Group

The command removes/deletes the protection group of type as specified by means of parameter protype. If protype equals 'EQPTEP51', then one instance of EP51 equipment protection group is deleted. The deletion of the protection group is reported by means of an DLT-PROTN-GRP notification.

EP51 equipment protection group - port impact due to EP51 protection group deletion

36 DS3 ports related to the protection pack of the protection group is created with default values. This creation is reported by means of a set of **ENT-T3** notifications

EP51 equipment protection group - tributary creation due to EP51 protection group deletion

Tributaries related to the ports of the protection pack of the protection group is created with default values. This creation is reported by means of a set of ENT-PTF-STS notifications

General transmission protection group - 1+1, 2F protection group

The command removes/deletes the protection group of type as specified by means of parameter protype. If *protype* equals ’1+1’, then one instance of 1+1 MS/line protection group is deleted. If *protype* equals ’2F’, then one instance of type 2-fibre MS-SPRing/BLSR protection group will be deleted.

The deletion of the protection group is reported by means of an DLT-PROTN-GRP notification.

General transmission protection group - 1:1 protection group

The command removes/deletes the protection group of type as specified by means of parameter protype. If protype equals ’1xNOPT’, then one instance of 1:1 MS/line protection group is deleted.

The deletion of the protection group is reported by means of an DLT-PROTN-GRP notification.

General transmission protection group - 4F protection group

The command removes/deletes the protection group of type as specified by means of parameter protype. If *protype* equals 4F’, then one instance of type 4 fibre MS-SPRing/BLSR protection group is deleted.

The deletion of the protection group is reported by means of an DLT-PROTN-GRP notification.

Multiplex section/Line protection group - DCC protection as part of 1+1 multiplex section/line protection

If DCC protection is enabled for section/rs dcc channels (value of dccsrsprotmd equals ENABLE), then following updates for DCC channels are done.

1. The dcc channel of type ’SECTION’ associated with the protection port is created This update is reported by means of a ENT-FECOM notification.
2. The protection mode (attribute dccprottp) of the dcc channel of type ’SECTION’ associated with the worker port is set to ’0X1’ unprotected. This update is reported by means of a ENT-FECOM notification for the changed attribute.

If DCC protection is enabled for line/ms dcc channels (value of dcclmsprotmd equals ENABLE), then following updates for DCC channels are done.

1. The dcc channel of type ’LINE’ associated with the protection port is created. This update is reported by means of a ENT-FECOM notification.
2. The protection mode (attribute dccprottp) of the dcc channel of type ’LINE’ associated with the worker port is set to ’0X1’ unprotected. This update is reported by means of a ENT-FECOM notification for the changed attribute.

Multiplex section/Line protection group - tributary impact due to 1+1 protection group - port redefinition due to 1+1 MSP protection group deletion

Tributaries signal rates of the protection port is set to default values (parameters inputsig, unequiposig), because they had been deleted at the enter time of the MSP 1+1 protection group.

As a side effect, the value of parameter crstat may also change if the port is operating in fixed mode

The update of the mentioned parameters are reported by means of:

- a ED-STM<n> notification if the protection port is an SDH port
- a ED-OC<n> notification if the protection port is an SONET port

The value for <n> is derived from the port rate.

Multiplex section/Line protection group - tributary impact due to 1+1 protection group - tributary creation due to 1+1 MSP protection group deletion

Tributaries related to the protection port of the protection group is recreated with default values, because they had been deleted at the enter time of the MSP 1+1 protection group.

This creation is reported by means of:

- a set of ENT-VC3 notifications if the protection port is an SDH port
- a set of ENT-STS1 notifications if the protection port is an SONET port

The number of tributaries to be created is derived from the port rate.

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to [2] 1&&-<N>, with N is maximal legal STS-1/AU-3 number .

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

2/4 fibre MS-SPRING/BLSR protection group

If the related ports are of portClass = EDGE any existing information in the loca, locz parameters of the related tributaries are deleted (i.e. set to the default value). This are reported by means of an ED-STS1 (istd=SONET) or ED-VC3 (istd=SDH) notification.

Example input/output

Equipment protection group

The following example deletes EP155 equipment protection group:

```
DLT-PROTN-GRP:LT-UNITE:1-1-eep01:123456::EQPTEP155;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
```

```
;
```

Multiplex section/line protection group 1+1

The following example deletes 1+1 multiplex section/line protection group:

```
DLT-PROTN-GRP:LT-UNITE:1-1-o001:1+1:123456::1+1;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

2 fibre MS-SPRing/BLSR protection group

The following example deletes 2F MS-SPRing/BLSR protection group:

```
DLT-PROTN-GRP:LT-UNITE:1-1-t001:123456::2F;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

4 fibre MS-SPRing/BLSR protection group

The following example deletes 4F MS-SPRing/BLSR protection group:

```
DLT-PROTN-GRP:LT-UNITE:1-1-f001:123456::4F;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses (LOXC equipment protection group)

Validation rule: protection unit is active and LO transmission entities

If the protecting unit is the active unit and low-order tributaries still exist, then the following error response is returned.

Error code:

`SNVS – Status, Not in Valid State, low-order transmission entities present`

It does not matter how the low-order transmission entities have been created (either implicitly by creating low-order cross-connections or by substructuring a high-order tributary). Even if no low-order cross-connections exist, low-order transmission entities may exist due to prior substructuring. The latter case will be corrected by deletion of substructures (**ED-SUBS** command) or low-order cross-connections must be deleted (**DLT-CRS** command).

In any case it is the better advise to activate the worker LOXC pack since the low-order transmission entities need not to be removed in this case.

Error responses (TransMUX equipment protection group)

Validation rule: protection unit is active and transmission entities exist

If the protection unit is active, and HO/LO cross-connections or cross-connection loopbacks or test access sessions exist which involve tributaries located at TransMUX packs in the protection group, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, protection unit is active`

Error responses (ES64U equipment protection group)

Validation rule: protection unit is active and transmission entities exist

If the protection unit is active, and HO cross-connections or cross-connection loopbacks or test access sessions exist which involve tributaries located at ES64U packs in the protection group, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, protection unit is active`

Error responses (general)

Validation rule: no test access session active

Tributaries related to ports being a member of the specified protection group must not be used in a test access session either as tributaries under test or as test access tributaries.

Error code:
`SNVS – Status, Not in Valid State,tributary used in test access session`

Error responses (2-fibre MS-SPRing/BLSR protection group)

Validation rule: consistent *aid* and *rid*

If both an *aid* and a *rid* are specified as input parameters and they are inconsistent, then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: No HO/LO cross-connection or LO substructure at tributaries

There shall exist no HO or LO cross-connection or LO substructure on tributaries related to ports being a member of the specified protection group.

Error code:
`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

Validation rule: no XC loopback on tribs

There shall exist no cross-connection loopbacks on tributaries related to ports being a member of the specified protection group:

Error code:
`SNVS – Status, Not in Valid State, Tributary is used in cross-connection loopback`

Error responses (4-fibre MS-SPRing/BLSR protection group)

Validation rule: consistent aid and rid

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: No HO/LO cross-connection or LO substructure at tributaries

There shall exist no HO or LO cross-connection or LO substructure on tributaries related to ports being a member of the specified protection group.

Error code:
`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

This validation includes coss connections and LO substructure at worker and protection ports, i.e. extra traffic cross-connections.

Validation rule: no XC loopback on tribs

There shall exist no cross-connection loopbacks on tributaries related to ports being a member of the specified protection group:

Error code:
`SNVS – Status, Not in Valid State, Tributary is used in cross-connection loopback`

Related TL1 messages

- "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
- "ED-PROTN-GRP: Edit protection group" (p. 3-803)
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
- "DLT-CRS: Delete cross-connection" (p. 3-145)
- "OPR-PROTNSW: Operate protection switch" (p. 3-2057)
- "RLS-PROTNSW: Release protection switch" (p. 3-2203)

DLT-PROTN-GRP notification

Output format

Command syntax: **DLT-PROTN-GRP**:*aid*:*protype*;

Output parameters

Command parameters (1+1 equipment protection group)

1. **aid** – Access Identifier
   This parameter specifies the aid of the 1+1 equipment protection group.
   Type: AID
   Range: EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID.
2. **protype** – Protection Type
   The parameter specifies the type of the protection group.
   Type: Alphanumeric value set.
   Range:
   - EQPTSWFBR (if *aid* is SwitchAndTmgProtGrpAID)
   - EQPTCTL (if *aid* is CtlProtGroupAID)
   - EQPTEP155 (if *aid* is EP155ProtGroupAID)
   - EQPTEP51 (if *aid* is EP51ProtGroupAID).
   - EQPTLOXC (if *aid* is LOXCProtGroupAID)
   - EQPTTMUX (if *aid* is TMUXProtGroupAID)
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

Command parameters (multiplex section/line protection group)

1. **aid** – Access Identifier
   This parameter specifies:
   - the aid of a 1+1 multiplex section/line protection group or
   - the aid of a 1:1 multiplex section/line protection group (only R3.0 and later).
   Type: AID.
   Range: OnePlusOne_ProtGrpAID or NByOne_ProtGrpAID.
2. **protype** – Protection Type
   The parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1x***N***OPT (only R3.0 and later).
   In mind the ***N*** in “1x***N***OPT” is considered to be always equal “1”.
   Type: Alphanumeric value set.
   Range: 1+1 (default) or 1xNOPT.
   Default: 1+1

Command parameters (2-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 2 fibre MS-SPRing/BLSR protection group.
   Type: AID.
   Range: TwoFibre_ProtGrpAID.
2. **`protype`** – Protection Type
   This parameter specifies the type of the protection group. The value for the 2-fibre MS-SPRing/BLSR protection group is 2F.
   Type: Alphanumeric value set.
   Range: 2F (default).

Command parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 4 fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: FourFibre_ProtGrpAID
2. **`protype`** – Protection Type
   This parameter specifies the type of the protection group. The value for the 4-fibre MS-SPRing/BLSR protection group is 4F.
   Type: Alphanumeric value set.
   Range: 4F.

Command parameters (Path protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the path protection group.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
   The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.
2. **`protype`** – Protection Type
   Thise parameter specifies the type of the protection group. The value for the path protection group is PATHDRI.
   Type: Alphanumeric value set.
   Range: PATHDRI.

Example output

1+1 Equipment protection group

The following example reports the deletion of an 1+1 equipment protection group:

```
   LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ”TIME=15-00-00,DATE=00-11-03:DLT-PROTN-GRP:1-1-estgrp:EQPTSWFBR”
;
```

Multiplex section/line protection group (1+1)

The following example reports the deletion of an 1+1 multiplex section/line protection group:

```
   LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:DLT-PROTN-GRP:1-1-o001:1+1″
;
```

2 fibre MS-SPRing/BLSR protection group

The following example reports the deletion of an 2F MS-SPRing/BLSR protection group:

```
   LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:DLT-PROTN-GRP:1-1-t001:2F″
;
```

4 fibre MS-SPRing/BLSR protection group

The following example reports the deletion of an 4F MS-SPRing/BLSR protection group:

```
   LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:DLT-PROTN-GRP:1-1-f001:4F″
;
```

Path protection group

The following example reports the deletion of a path protection group:

```
   LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:DLT-PROTN-GRP:1-1-#-#-10-1-25:PATHDRI″
;
```

DLT-PTF notification

Output format

Syntax

`DLT-PTF`:*aid;*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The access identifier specifies the PTF tributary whose attributes are reported.
   Type: AID
   Range:
   - PTNum for modifier: STS, HOSTS1, HOVC4, ALL
   - TMuxAPTNumAID for modifier: TMASTS1, ALL
   - TMuxSPTNumAID for modifier: TMSSTS1, ALL

   as defined in AID structure with exception of value *all*.
   Default: None.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-PTF:1-1-#-#-21-1-1"
;
```

DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-RC-NEIGHBOR**:*tid*::*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **remrcid** – Remote Routing Controller Identifier
   This parameter is used to indicate the Id of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.

Output format

If the **DLT-RC-NEIGHBOR** command is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″[rate]:thru[,lpaid]:spec_block″
   ...
   ″[rate]:thru[,lpaid]:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Detailed behavior description

The **DLT-RC-NEIGHBOR** command removes a routing controller from the list of neighboring routing controllers.

Example input/output

The following example shows the result of a DLT-RC-NEIGHBOR command:

```
RTRV-MAP-NEIGHBOR:LT-UNITE::123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   "STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=SECTION"
   "STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=LINE"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **DLT-RC-NEIGHBOR** command.

Validation rule: RC must be created

If the RC is not created (via ACT-L1ROUTING), then the command is denied.

Error code: `IDNV – Input, Data Not Valid, RC node not created`

Validation rule: RC must be in Neighboring List

The command is denied if the routing controller identified by remrcid is not in the neighboring RC list.

Error code: `IDNV – Input, Data Not Valid, RC unknown`

Related TL1 messages

- "RTRV-RC-NEIGHBOR: Retrieve all neighboring routing controllers" (p. 3-2816)
- "ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)

DLT-RC-NEIGHBOR notification

Output format

The DLT-RC-NEIGHBOR notification has the following specific syntax:

DLT-RC-NEIGHBOR:::*spec_block*

Output parameters

Spec block parameters

1. **remrcid** – Remote Routing Controller Identifier
   This parameter is used to indicate the Id of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
2. **remrcscn** – Remote Routing Controller SCN IP Address
   This parameter is used to indicate the IP address of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
3. **remrcraid** – Remote Routing Controller Routing Aread ID
   This parameter is used to indicate the routing area ID of the level 0 area of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   0.0.0.0 is not allowed.

Example output

```
RTRV-MAP-NEIGHBOR:LT-UNITE::123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F800000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=SECTION″
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F800000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=LINE″
;
```

DLT-STM1 notification

Output format

Syntax: **`DLT-STM1`**:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: SPortAID of STM1 port.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STM1:1-1-#-#-4-1″
;
```

DLT- STM1E notification

Output format

The DLT-STM1E notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Syntax: **`DLT-STM1`***:aid*

Output parameter

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: SPortAID of STM1E port

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STM1E:1-1-#-#-4-1″
;
```

DLT-STM4 notification

Output format

Syntax: **`DLT-STM4`**:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: SPortAID of STM4 port.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-STM4:1-1-#-#-4-1”
;
```

ED-STM16T autonomous notification

LOS for transmission port STM16T clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-STM16T notification with parameter *pmode*.

DLT-STM16 notification

Output format

Syntax: **`DLT-STM16`**:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: SPortAID of STM-16 port. The keyword **`all`** is not allowed.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-STM16:1-1-#-#-4-1”
;
```

DLT-STM16T notification

Output format

Syntax

`DLT-STM16T`:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **`all`** is not allowed.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STM16T:1-1-#-#-4-1″
;
```

DLT-STM64 notification

Output format

The DLT-STM64 notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **DLT-STM64**:*aid*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: STM-64 port AID.

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STM64:1-1-#-#-4-1″
;
```

DLT-STS1 notification

Output format

Syntax **`DLT-STS1`**:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   AID of the SONET tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  612 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STS1:1-1-#-#-4-1-1&&-768″
;
```

DLT-T3 notification

Output format

Syntax

`DLT-T3`:*aid*

Output parameters

Command parameters

1. **`aid`** – access identifier of a DS3 port
   Type: AID
   Range: APortAID of DS3 port
   The keyword **`all`** is allowed up to shelf level.
   Default: None.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:DLT-T3:1-1-#-#-21-1”
;
```

DLT-TCA-PROF: Delete TCA profile

General

The **DLT-TCA-PROF** command can be initiated by users to remove a Performance Monitoring (PM) threshold profile from the system memory. The system supports two default sets of profiles called ″Default″ and ″Default0″. The system default profiles cannot be deleted.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM3

Priority: 1

Abortable: No

Input format

Command syntax: **DLT-TCA-PROF**:*tid::ctag::pftype,pfname;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
4. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Output format

If the NE fully complies with the DLT-TCA-PROF request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

TCA Profile deletion

The command shall delete the existing performance monitoring Threshold Crossing Alert (TCA) profile.

The deletion of the TCA profile is reported by means of an DLT-TCA-PROF notification.

Example input/output

The following example shows the successful completion of a DLT-TCA-PROF command by the NE:

```
DLT-TCA-PROF:UNITE-NE::123456::section-line,att;
   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`DLT-TCA-PROF`** command.

Validation rule: profile name does not exist

If the profile name specified does not exist

Error code: `IDNV – Input, Data Not Valid, profile name does not exist`

Validation rule: deleting assigned profile

If a user requests to delete a profile name and there are ports or tributaries in the system that are provisioned to use the given profile for thresholding, the following error response is returned.

Error code: `SNVS – Status, Not in Valid State, TCA profile assigned`

Validation rule: deleting the system default profile

If a user requests to delete the system default profiles (Default and Default0), the following error response is returned.

Error code: `IDNC – Input, Data Not Consistent, default profile`

Related TL1 messages

- “RTRV-TCA-PROF: Retrieve TCA profile” (p. 3-2982)
- “ENT-TCA-PROF: Enter TCA Profile” (p. 3-1894)
- “ED-TCA-PROF: Edit TCA profile” (p. 3-1214).

References

GR-833-CORE

DLT-TCA-PROF notification

Output format

Command syntax: **`DLT-TCA-PROF`***::pftype,pfname*

Output parameters

Command parameters

1. **`pftype`** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
2. **`pfname`** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **`ENT-TCA-PROF`**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Example output

The following example shows the successful completion of a DLT-TCA-PROF command by the NE:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:DLT-TCA-PROF::RS-MS,DTAG″
;
```

DLT-TMAT3 notification

Output format

Command syntax: **DLT-TMAT3**::*aid*;

Output parameters

Command parameters

1. **aid** – Access Identifier of a TransMUX asynchronous adjunct port.
   Type: AID
   Range: TMuxAPortAID
   The keyword **all** is allowed up to shelf level.
   Default: None.

Example input/output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:DLT-TMAT3:1-1-#-#-21-tma1″
;
```

DLT-TMST3 notification

Output format

Syntax:

Command syntax: **`DLT-TMST3`**::*aid*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier of a TransMUX virtual synchronous adjunct port.
   Type: AID
   Range: TMuxSPortAID
   The keyword **`all`** is allowed up to shelf level.
   Default: None.

Example input/output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:DLT-TMST3:1-1-#-#-21-tms1"
;
```

DLT-ULSDCC-L3: Delete a node ID other than node-1

The command **DLT-ULSDCC-L3** is used to delete a node ID (osinode) in a UNITE NE which was created via ENT-ULSDCC-L3.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-ULSDCC-L3**:*tid*::*ctag*::*osinode*;

Input parameters

Command parameters

1. **tid** – Target Identifier
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
A 1675 LambdaUnite MSS NE has up to 8 instances of the OSI stack. Node-1 (osinode1) is default. Node 2-8 (osinode{2-8}) can be created and configured and deleted.
Type: AID
Range: osinode{2-8}.

Output format

If the DLT-ULSDCC-L3 is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command deletes a node ID (AID value of parameter osinode) at a NE. Only node IDs created via ENT-ULSDCC-L3 command can be deleted. The result should be an update of the OSI stack.

The command leads to a reset of the DCF controller and disconnection of management communication.

The update is reported by means of DLT-ULSDCC-L3 notification.

Example input/output

The following example shows the result of DLT-ULSDCC-L3:

```
DLT-ULSDCC-L3:NE-UNITE::0962415::osinode2;
   NE-UNITE 01-02-15 09:11:51
M  4744843 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the DLT-ULSDCC-L3 command.

Validation rule: Reject deletion when node is assigned to LAN or DCC

The command is rejected when a node is disabled to which DCC and/or LAN channel are assigned (via **ENT-FECOM**/**ENT-FECOM-LAN**).

Error code: `SNVS – Status, Not in Valid State, DCC/LAN channels assigned`

Validation rule: Reject deletion when node is used as RA destination node by OSI node 1

The command is rejected when a node is used as RA destination node by OSI node 1 (l3radesnode = 0).

Error code:

`SNVS – Status, Not in Valid State, node in use as RA destination node`

Validation rule: osinode must be in the created state

The osinode parameter for which changes should be applied must be in the created state, i.e. the value must be either osinode1 or the node must be successfully created via **ENT-ULSDCC-L3**.

Error code:

`SNVS – Status, Not in Valid State, osinode not in created state`

Related TL1 messages

- "RTRV-ULSDCC-L3: Retrieve layer 3 related parameters" (p. 3-3049)
- "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
- "ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)

DLT-ULSDCC-L3 notification

Output format

The DLT-ULSDCC-L3 notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The DLT-ULSDCC-L3 notification has the following specific syntax:

`DLT-ULSDCC-L3`::*osinode*

Output parameters

Command parameters

1. **`osinode`**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured and deleted.
   Type: AID.
   Range: osinode{2-8}.

Example output

```
   NE-UNITE 01-02-15 09:18:27
A  000 REPT DBCHG
   ”TIME=09-27-36,DATE=01-02-15:DLT-ULSDCC-L3::osinode8”
;
```

DLT-ULSDCC-L4: Delete TARP manual adjacent NSAP address in layer 4 of the specified OSI stack

The command **DLT-ULSDCC-L4** is used to delete one TARP manual adjacent NSAP address in the specified OSI node. The address to be deleted is specified by the system ID in the *spec_block* part of the command line.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-ULSDCC-L4**:*tid*::*ctag*::*osinode*:[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured and deleted.
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. l4ajsys – New Manual Adjacency: NET System Identifier
   This system parameter of the NSAP is set by the Equipment Manufacturer. The value is fix and globally unique among all NEs. For values of NSAP/NET system parameter.
   Type: Hexadecimal number.
   Range: 000000000000 – FFFFFFFFFFFF.

Output format

If the DLT-ULSDCC-L4 is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command deletes one NSAP address from the TARP Manual Adjacency list in layer 4 of a OSI stack. Result should be an update of the OSI stack. Multiple Manual Adjacent NEs are deleted using multiple occurrences of the DLT-ULSDCC-L4 command.

The update is reported by means of DLT-ULSDCC-L4 notification.

Example input/output

The following example shows the result of DLT-ULSDCC-L4:

```
DLT-ULSDCC-L4:NE-UNITE::4744843::osinode1:l4ajsys=08006A112345;
   NE-UNITE 01-01-18 10:09:52
M  4744843 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the DLT-ULSDCC-L4 command.

Validation rule: Check for existing TARP manual adjacent NSAP

The removal of a TARP manual adjacency is only allowed if an entry with the specified TARP manual adjacent NSAP exists for the *osinode*.

Error code:

`SNVS – Status, Not in Valid State, unknown TARP Manual Adjacent NSAP`

Related TL1 messages

- "RTRV-ULSDCC-L4: Retrieve OSI layer 4 related parameters" (p. 3-3055)
- "ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)

DLT-ULSDCC-L4 notification

Output format

The DLT-ULSDCC-L4 notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

The DLT-ULSDCC-L4 notification has the following specific syntax:

DLT-ULSDCC-L4::*osinode*:*spec_block*

Output parameters

Spec block parameters

1. **l4ajsys** – New Manual Adjacency: NET System Identifier
   This system parameter of the NSAP is set by the Equipment Manufacturer. The value is fix and globally unique among all NEs. For values of NSAP/NET system parameter.
   Type: Hexadecimal number.
   Range: 000000000000 – FFFFFFFFFFFF.

Example output

```
   NE-UNITE 01-01-19 04:29:01
A  000 REPT DBCHG
   "TIME=04-29-00,DATE=01-01-19:DLT-ULSDCC-L4::osinode1:l4ajsys=08006A112345"
;
```

DLT-USER-SECU: Delete user security

The **DLT-USER-SECU** command is used by an administrator to remove a non-administrator type user from the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No.

Input format

Command syntax: **DLT-USER-SECU**:*tid*:*uid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **uid** – User ID
   This parameter specifies the *uid* which should be deleted. Only a single uid is supported in the command.
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of uid values for the **ACT-USER** command are not allowed. Valid uid values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in theNE.
   Type: Alphanumeric string (case-sensitive).
   Range: 5 – 10 characters. The uid ALL is not allowed in any combination of upper or lower cases..
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

After receiving the **DLT-USER-SECU** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Description: The **DLT-USER-SECU** command is used by an appropriate administrator to delete a user id specified by *uid* from the NE. The **DLT-USER-SECU** command is applicable where a user account has to be deleted from the NE. The user data stored in the NE will be updated.

It will be reported via the corresponding "DLT-USER-SECU notification" (p. 3-275) section.

It is not possible to delete the user id of one of the predefined super-users.

NOTE: If the user with the specified *uid* is currently logged into the NE when this command is executed, the user session identified by this *uid* is **not** terminated. That means the user can continue to execute TL1 commands. To terminate the session manually, the "CANC-USER-SECU: Cancel user security" (p. 3-62) command has to be used.

Example input/output

The following example shows the successful completion of the **DLT-USER-SECU** command by the NE:

DLT-USER-SECU:LT-UNITE-123:nbguser1:123456;

```
   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Invalid *uid*

If the NE receives a DLT-USER-SECU command with a invalid (unknown) *uid*.

Error code: `IDNV – Input, Data Not Valid, invalid UID`

Validation rule: Attempt to delete administrator's *uid*

Attempt to delete administrator's *uid*.

Error code:

`IDNV – Input, Data Not Valid. Attempting to delete an administrator login. Two administrator user logins are required at all times`

Related TL1 messages

- "ENT-USER-SECU: Enter user security" (p. 3-1970)
- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

DLT-USER-SECU notification

Output format

The **DLT-USER-SECU** notification has the following specific syntax:

DLT-USER-SECU:*uid*

Output parameters

Command parameters

1. **uid** – User ID
This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of uid values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
Type: Alphanumeric string (case-sensitive).
Range: 5 – 10 characters. The uid ALL is not allowed in any combination of upper or lower cases.

Example output

The example shows a normal notification after an DLT-USER-SECU command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:DLT-USER-SECU:nbguser1”
;
```

DLT-VC3 notification

Output format

Syntax: **DLT-VC3**:*aid*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Description: AID of the SDH tributary. The keyword **all** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  612 REPT DBCHG
   "Time=8-00-30,DATE=00-11-20:DLT-VC3:1-1-#-#-4-1-1&&-768"
;
```

DLT-VCG – Enter virtual concatenation group

The **DLT-VCG** command provisions a virtual concatenation group of an ES64U together with all VCGTRIBs belonging to it.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

DLT-VCG-*modifier:tid:aid:ctag;*

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the virtual concatenation group of the unit. Implicit, it indicates the transmission rate for which VCG related data will be retrieved.
   The parameter *modifier* must be given if a virtual concatenation group on a GE10PL1 or ES64U pack shall be addressed. Otherwise this parameter has to be omitted.
   Type: Alphanumeric value set.
   Range:
   - 1GEPL representing a 1 GE port on the GE10PL1 pack
   - 10GEPL representing a 10 GE port on the GE10PL1 pack
   - ES64U representing a VCG port at an ES64U subsystem

   Default: None.
   The command is only allowed for *modifier* = ES64U.
2. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   The access identifier specifies the virtual concatenation group (VCG) port number to be deleted.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values
   - **vall** for Simple_VCGAID
   - **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID

- **`all`** for Simple_ShelfAID and
- **`all`** for BayAID

are excluded.
Default: None.

4. **`ctag`** – Correlation Tag
This parameter is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Type: Alphanumeric string

Range: non-zero decimal number or identifier of up to 6 characters

Note: Current implementation also allows '0' as valid CTAG, which is an extension to GR-831.

Default: None

Output format

If the network element fully complies with the **`DLT-VCG`** request, the following normal completion response shall be returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`DLT-VCG`** command generates a **`REPT DBCHG`** message for the deletion of the VCG together with a **`REPT DBCHG`** message for the deletion of the VCG tributaries (**`DLT-VCGTRIB`** notification).

The command delete a virtual concatenation group together with the virtual concatenation group tributaries belonging to it according to the parameters given with this command. The virtual concatenation group tributary number range is from 1 up to the number of tributaries given by the parameter *inputsig*.

The modification is reported by means of an **`DLT-VCG`** notification and a **`DLT-VCGTRIB`** notification.

Example input/output

```
DLT-VCG-ES64U:UNITE-NE:1-1-#-#-22-v9:123456;

  UNITE-NE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-VCG:1-1-#-#-22-v9"
;
  UNITE-NE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:DLT-VCGTRIB:1-1-#-#-22-v9-1&&-12"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`DLT-VCG`** command.

Validation rule: Command only allowed for VCG on ES64U circuit pack

If the command is not applied to a VCG on a ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: no crossconnection

The command is denied if there is still a cross-connection applied to any tributary of the whole VCG port. Whole port means all tributaries of a port independent of their configured membership.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: no loopbacks

The command is denied when a cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Related TL1 messages

- "DLT-VCG – Enter virtual concatenation group" (p. 3-277)
- "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)
- "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
- "ENT-VCG – Enter virtual concatenation group" (p. 3-1983)

DLT-VCG notification

Output format

The **DLT-VCG** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **DLT-VCG** notification has the following specific syntax:

DLT-VCG:*aid;*

Output parameters

Command parameter

aid – Access Identifier

This parameter specifies the VCG to be operated on.

Type: AID.

Range: VCGAID as defined in AID structure. The values

- *vall* for Simple_VCGAID
- *usall* for Simple_UnivSlotAID
- *all* for Simple_UnivSlotAID
- *all* for Simple_ShelfAID and
- *all* for BayAID

is excluded.

Example output

```
   UNITE-NE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:DLT-VCG:1-1-#-#-22-v1″
;
```

DLT-VCG-TRANS notification

Output format

The **DLT-VCG-TRANS** notification uses the general format of a REPT DBCHG notification.

Syntax

DLT-VCG-TRANS*:aid;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure with the exception of the value *vall*.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:DLT-VCG-TRANS:1-1-#-#-1-V1″
;
```

DLT-VCGTRIB notification

Output format

The **DLT-VCGTRIB** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **DLT-VCGTRIB** notification has the following specific syntax:

DLT-VCGTRIB:*aid;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG tributary whose attributes are reported.
   Type: AID.
   Range: VCGNumAID as defined in AID structure with exception of value all.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:DLT-VCGTRIB:1-1-#-#-1-V1-1&&-21”
;
```

DLT-VRTSW notification

Output format

The **`DLT-VRTSW`** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Command syntax: **`DLT-VRTSW`***:aid:vrtswid;*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The access identifier specifies the slot containing a GE1 pack.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure.
   The values
   - *all* and
   - *usall* for Simple_UnivSlotAID

   are accepted.
2. **`vrtswid`** – Virtual Switch Identifier
   This parameter identifiers the instance of the virtual switch.
   - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
   - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

   Type: Integer.
   Range: 1 – Currently this is the only allowed value.
   Default: 1.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:DLT-VRTSW:1-1-#-#-1:1″
;
```

ED-ASAP-PROF: Edit alarm severity assignment profile

The **`ED-ASAP-PROF`** command is used to change the name or parameter values for an ASAP.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **`ED-ASAP-PROF`**:*tid*::*ctag*::*pftype*,*pfname*:*spec_block*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`ctag`** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`pftype`** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen ("-") is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN

- SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

4. **`pfname`** – ASAP Profile Name
   This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
   Range: length 1-24 characters. For predefined ASAPs is the value of pfname “DEFAULT”, but it is no default name in general.

Spec block parameters

1. **`newname`** – New Profile Name
   This parameter specifies the new ASAP name of the ASAP specified by *pftype* and *pfname*. The name of any default ASAP can not be changed.
   See definition of *pfname* in the input parameters above
2. Alarm Severity Profile parameters
   The spec block contains ASAP parameters. Each ASAP parameter defines for a probable cause (condtype) the assigned severity which will be reported when the alarm appears. If for one specific probable cause a service affecting (sa) or non service affecting (nsa) alarm can be reported then for each the sa-alarm and nsa-alarm an ASAP parameter exists.
   The severity indicates the urgency of an alarm. The severity of a sa-alarm must be more urgent that of a nsa-alarm for the same condtype.
   The description of the ASAP parameters, the relation to the profile types and the default values are described in: “Alarms, events, ASAPs” (6-4).
   All parameters have the same type and value range. Range: · · · · ·
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)
   - MN – Minor (ITU / Telcordia)
   - NA – Warning (ITU) / Not-Alarmed (Telcordia)
   - NR – Non-Alarmed (ITU) / Not-Reported (Telcordia).

   Example: Concrete ASAP parameters are *sa_eqpt* and *nsa_eqpt*. Both refers to the probable cause (*condtype*) EQPT and ASAP type EQUIPPROT. *sa_eqpt* contains the assigned severity for the service affecting alarm of probable cause EQPT and *nsa_eqpt* for the non service affecting alarm. *sa_eqpt* can have the concrete value CR and *nsa_eqpt* MN.

Output format

The following example shows the successful completion of the **ED-ASAP-PROF** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

If *newname* is specified the ASAP name will be changed to this new profile name. The specified ASAP parameters will be updated to its new severities.

If a ASAP parameter is not specified as part of the *spec_block*, then its value remains unchanged.

The default ASAP parameter values of the profiles named DEFAULT can be modified.

The modification is reported via the ED-ASAP-PROF notification.

Note: ED-ASAP-PROF notification includes all parameters instead of only changed parameters.

Update affected entities

The new ASAP name (if specified) will be updated in all ASAP pointer attributes of other entities of the system. Notice: No **REPT DBCHG** message will be sent reporting this change.

Note: There will be notifications generated for the affected entities.

Example input/output

The following example shows the successful completion of the **ED-ASAP-PROF** command by the NE if the *pfname* will be changed:

```
ED-ASAP-PROF:LT-UNITE-123::123456:PORT,NUERNBERG:newname=CHEMNITZ;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

The following example shows the successful completion of the ED_ASAP-PROF command by the NE if assigned severities will be changed:

```
ED-ASAP-PROF:LT-UNITE-123::123456::PORT,NUERNBERG:sa_los=MJ,sa_lof=MJ,sa_eber=MJ;

   LT-UNITE-123 00-01-01 08:00:00
```

```
M  123456 COMPLD
;
```

The following example shows the successful completion of the ED_ASAP-PROF command by the NE if assigned severities and the ASAP name will be changed:

```
ED-ASAP-PROF:LT-UNITE-123::123456::PORT,NUERNBERG:newname= CHEMNITZ,sa_los=MJ,sa_
      lof=MJ,sa_eber=MJ;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`ED-ASAP-PROF`** command.

Validation rule: *pfname* does not exist

An ASAP with the name *pfname* does not exist.

Error code:

`SDNC – Status, Data Not Consistent, invalid instance of profile`

Validation rule: Rename of default ASAP

A new profile name is specified for a default profile.

Error code:

`IDNC – Input, Data Not Consistent, new name for default profile`

Validation rule: Inconsistent severity level

An ASAP parameter value has severity level inconsistency (for example, the Non Service Affecting value of a parameter is CR and the Service Affecting value is MJ).

Error code:

`IDNC – Input, Data Not Consistent, NSA severity is higher than SA severity`

Validation rule: Duplicate *newname*

The *newname* is already used for the specified *pftype*.

Error code: `IEAE – Input, Entity Already Exists, duplicate profile name`

Related TL1 messages

- "DLT-ASAP-PROF: Delete alarm severity assignment profile" (p. 3-134)
- "ENT-ASAP-PROF: Enter alarm severity assignment profile" (p. 3-1446)

- "RTRV-ASAP-PROF: Retrieve ASAP profile" (p. 3-2249)
- "RTRV-ASAP-ASGNMT: Retrieve ASAP assignment" (p. 3-2244)

ED-ASAP-PROF notification

Output format

Syntax: The **ED-ASAP-PROF** notification has the following specific syntax:

ED-ASAP-PROF::*pftype*,*pfname*:*spec_bloc*

Output parameters

Command parameters

1. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
   - TIMING: System Timing
   - TRANSPARENCY: Fully Transparent Service

   Default Value: There is no default value. The *pftype* does not change.
2. **pfname** – Profile Name
   This is the profile name. It is the previous profile name in case the profile name was changed with *newname*.

Spec block parameters

1. **`newname`** – New Profile Name
   This parameter specifies the new ASAP name of the ASAP specified by *pftype* and *pfname*. The name of any default ASAP can not be changed.
   This indicates the name of the ASAP.
   The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
   Type: Alphanumeric character string A hyphen (“-”) is allowed in the name additionally.
   Range: length 1-24 characters
   Default: none. For predefined ASAPs is the value of pfname ″DEFAULT″, but it is no default name in general.
2. Alarm Severity Profile parameters
   The spec block contains ASAP parameters. Each ASAP parameter defines for a probable cause (condtype) the assigned severity which will be reported when the alarm appears. If for one specific probable cause a service affecting (sa) or non service affecting (nsa) alarm can be reported then for each the sa-alarm and nsa-alarm an ASAP parameter exists.
   The severity indicates the urgency of an alarm. The severity of a sa-alarm must be more urgent that of a nsa-alarm for the same condtype.
   The description of the ASAP parameters, the relation to the profile types and the default values are described in: “Alarms, events, ASAPs” (6-4).
   All parameters have the same type and value range. Range: · · · · ·
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)
   - MN – Minor (ITU / Telcordia)
   - NA – Warning (ITU) / Not-Alarmed (Telcordia)
   - NR – Non-Alarmed (ITU) / Not-Reported (Telcordia).

   Example: Concrete ASAP parameters are *sa_eqpt* and *nsa_eqpt*. Both refers to the probable cause (*condtype*) EQPT and ASAP type EQUIPPROT. *sa_eqpt* contains the assigned severity for the service affecting alarm of probable cause EQPT and *nsa_eqpt* for the non service affecting alarm. *sa_eqpt* can have the concrete value CR and *nsa_eqpt* MN.

Example output

The example shows a normal notification after an ED-ASAP-PROF command if the ASAP name was changed:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-ASAP-PROF::ENET,NUERNBERG:newname=CHEMNITZ″
```

```
;
```

The example shows a normal notification after an ED-ASAP-PROF command if severities was changed:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-ASAP-PROF::ENET,NUERNBERG:sa_vcg_loa=MJ,nsa_vcg_
      loa=MN,sa_vcg_ssf=MJ,nsa_vcg_ssf=MN″
;
```

The example shows a normal notification after an ED-ASAP-PROF command if the ASAP name and severities was changed:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-ASAP-PROF::ENET,NUERNBERG:newname=FUERTH,sa_vcg_
      loa=MJ,nsa_vcg_loa=MN,sa_vcg_ssf=MJ,nsa_vcg_ssf=MN″
;
```

ED-CRS: Edit cross-connection

The **ED-CRS**-modifier command can be initiated by a user to modify the following cross-connection parameters:

- *loca/locz* in specifying the source-/destination TID associated with *in_aid/out_aid*
- *omode* specifying the output-mode of the cross-connection
- *xcappl* specifying the cross-connection application number of the cross-connection
- *xcnum* specifying the cross-connection number of the cross-connection.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **ED-CRS**-*modifier*:*tid*:*in_aid*,*out_aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **ED-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48

- STS192
- AU3TU3
- TU3AU3

No **all** value is supported.

2. **tid** – Target Identifier
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
3. **in_aid** – Input Access Identifier
This parameter specifies the source of the cross-connection.
For a detailed description please refer to the in_aid input parameter in "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).
Type: GenTribAID or GenLOTribAID.
Range: No use of the **all** or **usall** or **vall** values within the *out_aid* is allowed.
4. **out_aid** – Output Access Identifier
This parameter specifies the tributary which is the sink of the cross-connection.
For a detailed description please refer to the out_aid output parameter in "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).
Type: GenTribAID or GenLOTribAID
Range: n.a. No use of the **all** or **usall** or **vall** values within the *out_aid* is allowed.
5. **ctag** – Correlation Tag
This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
Type: Alphanumeric string
Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **loca** – Location-a
This parameter specifies the TID of the node at which service is added to the BLSR.
This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections, if the Automatic Squelch Map Calculation function is disabled.
The parameter *loca* can be specified for BLSR THROUGH, INTER-BLSR and DROP connections while the Automatic Squelch Map Calculation function is enabled to do a user override. The *loca* parameter value has to be omitted for ADD, non-BLSR connections.
A 2-way cross-connection is considered to be a DROP if the *out_aid* is a non BLSR tributary.
For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring

node then *loca* is the TID of the node where the related HO circuit is first added to the ring. Refer to the figure "Lower-order cross-connections" (p. 3-2322) for further explanation.
At the lower-order side of a conversion cross-connection, the same rules for *loca* hall be applicable as for lower-order cross-connections.

Additional information are given for this parameter in the RTRV-CRS section with respect to:

- figure illustrating BLSR connections,
- inter-ring cross-connections with collapsed nodes,
- extra traffic handling.

Type: TID or empty string.
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string "":
Note: In output direction the empty string appears as \"\".

2. **locz** – Location-z

   This parameter specifies the TID of the node at which service is dropped from the BLSR.
   This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections, if the Automatic Squelch Map Calculation function is disabled.
   The parameter *locz* can be specified for BLSR THROUGH, INTER-BLSR and ADD connections while the Automatic Squelch Map Calculation function is enabled to do a User Override.
   The *locz* parameter value has to be omitted for DROP and non-BLSR connections.
   A 2-way cross-connection is considered to be a ADD if the *in_aid* is a non BLSR tributary.
   For lower-order cross-connections the parameter *locz* is the TID of that ring node in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides. If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring.
   Refer to the figure in "Lower-order cross-connections" (p. 3-2322) for further explanation.
   At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.

   Additional information are given for this parameter in the RTRV-CRS section with respect to:

   - figure illustrating BLSR connections,
   - inter-ring cross-connections with collapsed nodes,
   - extra traffic handling.

   Type: TID.
   Range: 1 to 20 characters. Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.

3. **`omode`** – Output Mode

   This parameter specifies the output mode of the cross-connection:

   - for a 1-way cross-connection the output is clearly defined by the destination (*out_aid*)
   - for a 2-way cross-connection the output is defined as the source-to-destination direction (*in_aid* to *out_aid*)
   - NORM: this is the normal output i.e. the signal from the *in_aid* is passed through to the *out_aid*;
   - IDLE/UNEQ: this mode outputs an unequipped signal;
   - AIS: this mode outputs an AIS signal.

   omode and adaptive-mode: In case a SONET-port is provisioned to adaptive-mode setting the *omode* to IDLE/UNEQ or AIS will be lead to transmitting the corresponding signal (UNEQ or AIS) on STS1 base for all the STS1 tributary comprising the STS-N signal. Thus for an output of an STS3 cross-connection, if either AIS or IDLE/UNEQ is selected each of the 3 STS1 will output the AIS or IDLE/UNEQ signal. When switching the *omode* back to NORM the original STS3 cross-connection-signal will re-appear again.
   omode in case of path-protected cross-connection: In case of a path-protected cross-connection the *omode* affects the output of the path selector and is thus not setable on a per leg base. In other words the *omode* insertion is done after (downstream direction) the selector function.
   Type: Alphanumeric value set.

   Range:

   - NORM (default)
   - IDLE/UNEQ
   - AIS

4. **`xcappl`** – cross-connection application

   This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies. The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.

   As a possible assignment proposal the following may serve:

   - 0) Unknown
   - 1) 1-Way Point-to-Point
   - 2) 1-Way Path-Protected
   - 3) 1-Way Adjunct Path-Protected
   - 4) 1-Way Conversion Cross-Connection
   - 10) 2-Way Point-to-Point
   - 11) 2-Way Conversion Cross-Connection

- 20) UPSR (or SNCP Ring) Add, Drop
- 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
- 22) UPSR (or SNCP) Drop, 1-Way Broadcast
- 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
- 30) Logical Ring (or SNCP) Add, Drop
- 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
- 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
- 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
- 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)
- 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
- 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

Type: numerical value set.
Range: {0, 1, 2,…, 255}.
Default: The default value is 0.

5. **`xcnum`** – cross-connection number
This parameter is a 12 digit number, identifying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. described in ″Example input/output″ section of the **`RTRV-CRS`** command like UPSR, DRI etc.). Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in ring interworking scenarios).
The following format is recommended:
two digits for the bay, one digit for the shelf, two digita for the slot, two digits for the port, three digits for the high-order tributary, and two digits for hte low-order tributary.
Type: numeric value set (12 digit).
Range: 000000000000 .. 999999999999.
Default: The default value is 000000000000.

Output format

If the **`ED-CRS`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

;

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of the "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

A cross-connection can be modified via the **ED-CRS** command when *xconcls* = TRADITIONAL or when only the parameters loca, locz are modified.. A privileged user may change *xconcls* to TRADITIONAL via ED-NNCRS.

The affected parameter, as listed in the *spec_block* parameters part of this command, will be updated in the corresponding cross-connection entity.

For updates in more complex scenarios the following holds: -update of *omode* of one leg for a path-protected XC will also update *omode* of the second leg – thus 2 ED-CRS notifications will be sent; -update of *locz* of one leg within a path-protected XC will also update *locz* of the second leg -update of *loca* of one leg will also update *loca* of all other XCs sharing *in_aid* as input.

Furthermore the *repleg* status will be updated to *loca/locz* change (conversion from 1-way to 2-way or vice versa).

Update *loca*/*locz* of low-order cross-connections

If the *loca*/*locz* information is changed for a low-order cross-connection at an MS-SPRING port, then *loca*/*locz* of all low-order cross-connections using the same HO PTF at the MS-SPRING port is updated to the same value.

Update *loca*/*locz* of conversion cross-connections

If the *loca*/*locz* information is changed for a conversion cross-connection at an MS-SPRing port, then *loca/locz* of all conversion cross-connections using the same HO PTF at the MS-SPRing port is updated to the same value.

Identification of VCG tributaries

The system shall allow to edit cross-connections at VCG tributaries using the AID format of an AU-3/STS-1 tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

Example input/output

Example: For an existing non-BLSR 1-way cross-connection with rate vc4 the omode will be set to AIS via:

```
ED-CRS-VC4:UNITE-NE:1-1-#-#-10-1-4,1-1-#-#-20-1-7:123456:::omode=AIS;


   UNITE-NE 00-11-14 12:00:00
```

```
M  123456 COMPLD
;
```

Error responses

Validation rule: Entities (tributaries) must exist

Cross-connection command processing verifies that the logical tributaries specified in the command do exist.

Error code:

`IENE – Input Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Entity (cross-connection) must exist

Cross-connection command processing verifies that the specified cross-connection exists.

Error code:

`IENE – Input Entity Not Exists, Specified cross-connection does not exist`

Validation rule: Specified rate consistent with the cross-connecton rate

Cross-connection command processing verifies that the specified rate (*modifier* parameter) is compliant with the rate of the existing cross-connection.

Error code:

`IPNC – Input Parameter Not Consistent, Rate inconsistent with cross-connection rate`

Validation rule: Only those parameter present which are needed

command processing shall verify according to the given context only those parameter are present which are needed for that context.

This means that e.g.

- for editing a non-BLSR or BLSR-ADD cross-connection the *loca* parameter must not be present
- for editing a non-BLSR or BLSR-DROP cross-connection the *locz* parameter must not be present
- for editing a non-BLSR and unprotected cross-connection the *loca* and *locz* parameter must not be present

Error code:

`IPNC – Input Parameter Not Consistent, Rate inconsistent with cross-connection rate`

Validation rule: TID validation

Cross-connection command processing verifies that the new TID(s) (*loca* and/or *locz*) are compliant with the actual cross-connection configuration according to the following rules:

- *loca* is a non-empty string in case the *in_aid* is involved in a BLSR protection roup and the automatic squelch map calculation is disabled
- *locz* is a non-empty string in case the *out_aid* is involved in a BLSR protection group and the automatic squelch map calculation is disabled

Error code:

`IPNC – Input Parameter Not Consistent, TID not compliant with cross-connection configuration`

Validation rule: Presence of spec_block parameter

Cross-connection command processing shall verify that at least one spec_block parameter is submitted with the command.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Cross-connection must not be of *xconcls* NN

Cross-connection command processing shall verify, that the cross-connection is not of *xconcls* NN if other parameters than *loca* and *locz* are changed..

Error code:

`SNVS – Status, Not in Valid State, Cross-connection must not be of xconcls NN`

Validation rule: loca/locz information only retrievable for EXTRA TRAFFIC

Cross-connection command processing shall deny commands, if affected tributaries belong to EXTRA TRAFFIC and loca/locz information is specified in the command. The loca/locz information is automatically created for EXTRA TRAFFIC by the NE.

Error code:

`IDNC – Input, Data Not Consistent, It is not allowed to specify loca/locz for EXTRA TRAFFIC`

Related TL1 messages

- "DLT-CRS: Delete cross-connection" (p. 3-145)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)
- "ENT-ROLL: Enter roll" (p. 3-1796)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "ED-NNCRS: Edit NN cross-connection" (p. 3-457)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)

References

GR-199-CORE

ED-CRS notification

Output format

Syntax: **ED-CRS**-*modifier*:*in_aid*,*out_aid*:[*in_aid2*]:*spec-block*;

Output parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **ED-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - AU3TU3
   - TU3AU3

   No **all** value is supported.
2. **in_aid** – Input Access Identifier
   This parameter specifies the source of the cross-connection.
   For a detailed description please refer to the in_aid input parameter in "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).
   Type: GenTribAID or GenLOTribAID.
   Range: n.a. No use of the **all** or **usall** or **vall** values within the in_aid is allowed.

3. **out_aid** – Output Access Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   Type: GenTribAID or GenLOTribAID. STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs, VCGNumAID are used for high-order tributaries on e.g. GE***n***, OPT2G5 packs, GenLOTribAID are used for low-order tributaries on I/O packs e.g. OP***n***, EP155.
   Range: n.a. (**all** or **usall** or **vall** values shall never occur as part of an output parameter).
4. **in_aid2** – Input Access Identifier 2
   This parameter is applicable to and is present for path-protected cross-connections only and specifies either the source of the protection leg or the source of the working leg.
   This depends on the value of the *repleg* parameter:
   - E.g. for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSW the worker leg is given by IN1 -> OUT and the protection leg is given by IN2 -> OUT;
   - for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSP the worker leg is given by IN2 -> OUT and the protection leg is given by IN1 -> OUT.

   Thus *in_aid* and *out_aid* specify one leg of the legpair of a path-protected cross-connection, while *in_aid2* and *out_aid* specify the second leg.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
   STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
   Note: Using VCGNumAID (when using e.g. GE***n*** packs) is NOT applicable as those packs can not be the input to a path-protected cross-connection. Therefore specific AID types are mentioned instead of GenTribAID.
   Range: n.a. (**all** values shall never occur as part of an output parameter).

Spec block parameters

1. **loca** – Location-a
   This parameter *loca* (location-a) specifies the TID of the node at which service is added to the BLSR. T
   his parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections. The *loca* parameter value is an empty string for ADD and non-BLSR connections.
   A 2way cross-connection is considered to be a DROP if the out_aid is a non BLSR tributary. The figure “2-way scenario” (p. 3-2322)in section RTRV-CRS (the 3 NEs should be seen as part of an BLSR) depicts the 2way scenario.
   Inter ring cross-connections with collapsed node: Note that beside the above described THROUGH case there can be also another situation where both *loca* and *locz* are different from the empty string: If two rings are interconnected in a collapsed node then *loca* refers to the (add-node of the) ring where the signal is dropped from, and *locz* refers to the (drop-node of the) ring where the signal is

added to. In such cases the cross-connection cannot be provisioned as bidirectional, thus the information is still unambiguous.

A note to the loca2 parameter: Although *loca2* (Location-a-2) is a parameter which is used in the **ENT-CRS** command, *loca2* is not retrieved explicitly. This is due to the fact that for path-protected cross-connections – and only for those this *loca2* is applicable – the retrieval will be on a leg-basis and not on a leg-pair basis. In other words: If a path-protected cross-connection is retrieved two entries will be sent, one for the working leg (incl. the *loca* and *locz* parameter) and one for the protection leg (incl. the *loca* parameter and the same *locz* parameter, and in this case the *loca* returned is the *loca2* as entered when establishing the path-protected cross-connection).

The parameter *loca* is also required for EXTRA TRAFFIC connections. In contrast to incoming references from other commands to this whole chapter it is exclusively applicable for the command **RTRV-CRS**.

For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of loca refer to the parameter *sqfa*.

For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring node then loca is the TID of the node where the related HO circuit is first added to the ring.

The picture “Lower-order cross-connections” (p. 3-2322)in section RTRV-CRS shows an example of a LO path through a part of a ring. The LO path is added to/dropped from the ring in NE-1 and NE-2. 2-way cross-connections are considered to be either an Add or a Drop connection depending whether the input or the output tributary is at the ring port.

In NE-3 there is a HO through connection, which contains the LO path. In NE-3 there is no HO path termination. In NE-4 there is a LO through connection. Consequently, there are HO path terminations in NE-4.

The fact that loca refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same loca value.
- Any LO through-connection will be treated as add/drop. loca will identify the node of the through connection, and not the node where the LO service is added to the ring.

At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

2. **locz** – Location-z
This parameter *locz* (Location-z) specifies the TID of the node at which service is dropped from the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections. The *locz* parameter value is an empty string for DROP and non-BLSR connections. A
2way cross-connection is considered to be a ADD if the in_aid is a non BLSR tributary.
Refer to figure “2-way scenario” (p. 3-2322) in thr RTRV-CRS section for further explanation.
locz is also required for EXTRA TRAFFIC connections.
For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of *locz* refer to the parameter *sqfz*.
For lower-order cross-connections the parameter *locz* is the TID of that ring node in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides. If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring. Refer to figure “Lower-order cross-connections” (p. 3-2322) in the RTRV-CRS section for further explanation.

The fact that *loca* refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same loca value.
- Any LO through-connection will be treated as add/drop. loca will identify the node of the through connection, and not the node where the LO service is added to the ring.

At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: "" Note: In output direction the empty string appears as \"\".

3. **omode** – Output Mode

   The parameter *omode* specifies the output mode of the crossconnection:

   - for a 1way cross-connection the output is clearly defined by the destination (*out_aid*)
   - for a 2way cross-connection the output is defined as the source-to-destination direction (*in_aid* to *out_aid*)

   NORM: this is the normal output i.e. the signal from the *in_aid* is passed through to the *out_aid*.
   IDLE/UNEQ: this mode outputs an unequipped signal.
   AIS: this mode outputs an AIS signal
   omode and adaptive-mode:
   In case a SONET-port is provisioned to adaptive-mode setting the *omode* to IDLE/UNEQ or AIS will be lead to transmitting the corresponding signal (UNEQ or AIS) on STS1 base for all the STS1 tributary comprising the STS-N signal. Thus for an output of an STS3 cross-connection, if either AIS or IDLE/UNEQ is selected each of the 3 STS1 will output the AIS or IDLE/UNEQ signal. When switching the *omode* back to NORM the original STS3 cross-connection-signal will re-appear again.
   omode in case of pathprotected cross-connection:
   In case of a pathprotected cross-connection the *omode* affects the output of the pathselector and is thus not setable on a per leg base. In other words the *omode* insertion is done after (downstream direction) the selector function.
   Type: Alphanumeric value set.

   Range:

   - NORM
   - IDLE/UNEQ
   - AIS

   Default: NORM

4. **`onnsConnControl`** – ONNS Connection Control
   This parameter indicates for cross-connections of *xconcls* NN, whether the ONNS connection a cross-connection is assigned to is controlled by ONNS directly or by an external control (setup via UNI or eNNI or as SPC connection).
   For cross-connections of *xconcls* TRADITIONAL the value UNKNOWN is reported for this parameter.
   The value UNKNOWN shall also be reported for cross-connections of *xconcls* NN, the onnsConnControl of which cannot be determined.
   Type: Alphanumeric value set.
   Range:
   - UNKNOWN
   - ONNS
   - EXT
5. **`onnsPathID`** – ONNS Path Identity
   This parameter indicates for cross-connections of *xconcls* NN the path ID of the ONNS path a cross-connection is assigned to.
   For cross-connections of *xconcls* TRADITIONAL an empty string is reported for this parameter.
   Type: Character string of length up to 20 or empty string
   Range:
   nodeID-Index
   - nodeID: four dot separated numeric strings: each string 0..255
   - Index: Numeric string 0-65535

   Example: 29.13.196.203-65111
   Default: empty string
6. **`repleg`** – Reported Leg Pair
   This parameter provides compact information about the leg(s) of cross-connections and is derived, as shown in the table below, from a combination of the following elementary parameter:
   - atomic cross-connection topology
   - leg status
   - path-protected group working leg (if applicable i.e. being part of a protection group).

Atomic cross-connection Topology	Path Protection Leg Role	Leg Status	Reported Legpair (*repleg*)
point-to-point	n.a.	1-way	1WAY
point-to-point	n.a.	2-way	2WAY
path-protected	W	1-way	1WAYPSW
path-protected	P	1-way	1WAYPSP

Atomic cross-connection topology:

Two atomic cross-connection topologies are supported:

- Point to point: The point to point cross-connection consists of one leg only, which is defined by its source-trib (*in_aid*) and its destination-trib (*out_aid*) and its particular transmission rate.
- path-protected: The path-protected cross-connection consists of a path protection group with two legs, a working leg and a protection leg – the declaration of working respectively protection is made when establishing the cross-connection (ENT-CRS command).

  The two legs are defined by two input tributaries and one output tributary – thus one input tributary and the output tributary form the working leg while the other input tributary and the output tributary form the protection leg.

  Note that:

  both input tributaries must belong to ports with the same interface standard (either SONET or SDH), due to different switch criteria;

  both legs must be at the same cross-connection rate.

Leg status:

The leg status indicates whether a leg is monitored and reported as a 1-way cross-connection or a leg pair i.e. a 2-way cross-connection.

1-way: The leg status for a path-protected cross-connection is always defined as 1-way.

2-way: The leg status for a (non path-protected) cross-connection is set to 2-way if all the following conditions are fulfilled: there are 2 cross-connections in opposite direction between the same tributaries; both of these cross-connections have the same rate (independent if the SONET or SDH rate was used when establishing i.e. VC3 is equivalent to STS1, VC4 is equivalent to STS3 etc); the source and destination TIDs do match i.e. the source TID of one leg is the destination TID of the other leg and vice versa (this condition of course holds only in case of BLSR cross-connections).

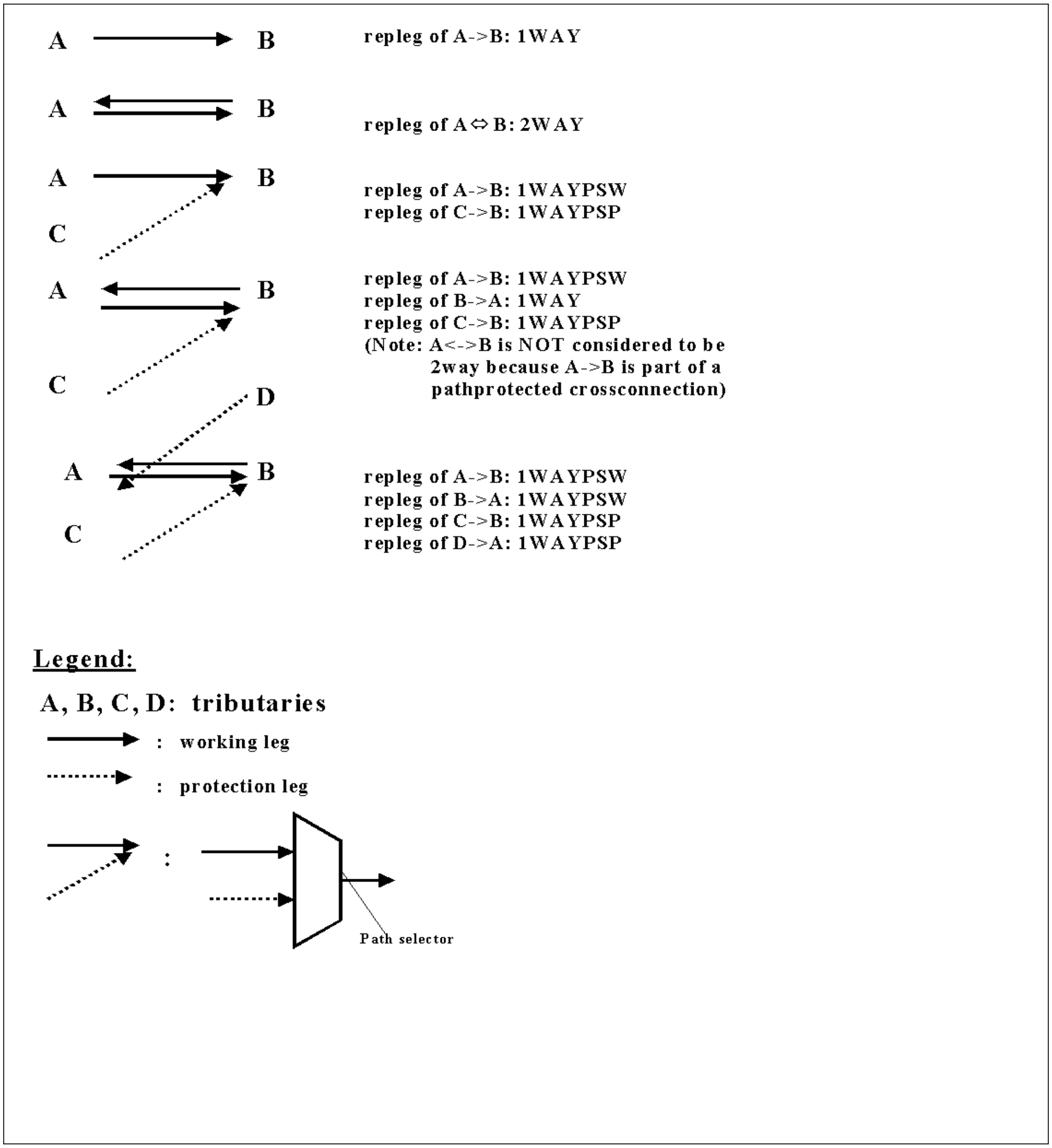
Note that the criteria for a 2-way leg-pair do not depend on whether the two legs were set up at the same time, nor on whether either leg was set up as part of a 2-way cross-connection topology

Path Protection leg role:

This indicates if the leg of a path protection group is defined as the working (W) leg or the protection (P) leg.

1WAYPSW and 1WAYPSP stands for 1-way path switched working and 1-way path switched protection respectively.

The following figure shows examples for the *repleg* parameter for several cross-connection configurations including compound cross-connection.



Range:

- 1WAY
- 2WAY
- 1WAYPSW
- 1WAYPSP.

The values 1WAYPAP and 1WAYPAW are not supported in the initial release.

7. **`rtntaccstat`** – Return Test Access Status
   This parameter indicates whether the input tributary of the return direction (destination-to-source) of a 2-way cross-connection is cross-connected or bridged to a test access tributary of a test session. In the case of a 1-way cross-connection the parameter is omitted.
   Type: Alphanumeric value set.
   Range: YES, NO (default).
8. **`taccstat`** – Test Access Status
   This parameter indicates whether the input tributary of a 1-way cross-connection or the input tributary of the source-to-destination direction of a 2-way cross-connection is cross-connected or bridged to a test access tributary of a test session.
   Type: Alphanumeric value set.
   Range: YES, NO (default).
9. **`xcappl`** – cross-connection application
   This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies. The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.

   As a possible assignment proposal the following may serve:

   - 0) Unknown
   - 1) 1-Way Point-to-Point
   - 2) 1-Way Path-Protected
   - 3) 1-Way Adjunct Path-Protected
   - 4) 1-Way Conversion Cross-Connection
   - 10) 2-Way Point-to-Point
   - 11) 2-Way Conversion Cross-Connection
   - 20) UPSR (or SNCP Ring) Add, Drop
   - 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
   - 22) UPSR (or SNCP) Drop, 1-Way Broadcast
   - 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
   - 30) Logical Ring (or SNCP) Add, Drop
   - 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
   - 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
   - 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
   - 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)

- 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
- 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

Type: numerical value set.
Range: {0, 1, 2,..., 255}.
Default: The default value is 0.

10. **`xcnum`** – cross-connection Number
Note: *rtnxcappl*, *rtnxcnum* and *rtnomode* are not listed because these parameters are always in the scope of one leg. Note that in contrast to CG the ED-CRS notification is used to notify *repleg* changes – CG therefore did use the ENT-CRS notification which is not logical.
The parameter *xcnum* (cross-connection number) is a multiple digit number, identifiying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. decribed in ″Example input/output″ section of the **`RTRV-CRS`** command like UPSR, DRI etc.).
Like the xcappl parameter the xcnum parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in Ring Interworking scenarios).
The following format for the 12 digits is recommended:
   - two digits for the bay
   - one digit for the shelf
   - two digits for the slot
   - two digit for the port
   - three digits for the high-order tributary and
   - two digits for the low-order tributary.

   If a high-order tributary is addressed then the last two digits are set to 00. The digits values pertain to the out_aid.
Type: numeric value set (12 digit)
Range: 000000000000 .. 999999999999
Default: The default value is 000000000000.

11. **`xconcls`** – cross-connection class
Note: Modification can only be done by a privileged user via ED-NNCRS.
This parameter identifies, whether the cross-connection has been setup in the TRADITIONAL way by an operator via the **`ENT-CRS`** command or via the NN application as part of an ENT-PATH command. This parameter therefore also indicates who is the owner of the cross-connection and can modify or delete it.
Note: Refer to the NN TL1 catalogue for the **`ENT-PATH`** command.
Type: Alphanumeric value set.
Range: TRADITIONAL, NN.

Example output

Refer to the ″Detailed behavior description″ of "ED-CRS: Edit cross-connection" (p. 3-292).

ED-DAT: Edit date and time

The **ED-DAT** command can be initiated by a user to change the date and time information in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: no

Input format

Command syntax: **ED-DAT**:*tid*::*ctag*::*[date][,time]*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **date** – Date
   The *date* specifies the current date as YY-MM-DD. If the parameter *time* is not specified, the current value is not changed, but then *date* must be specified for the **ED-DAT** command to complete successfully.
   Type: value set.
   Range: YY-MM-DD, with YY between 00 and 99, MM between 01 and 12, DD between 01 and 31.
   Default: 00-01-01 after start-up with default DB, current NE value, if not supplied in the command.
4. **time** – Time
   The *time* specifies the current time of day as HH-MM-SS. If the parameter *date* is not specified, the current value is not changed, but then *time* must be specified for the **ED-DAT** command to complete successfully.
   Type: value set.
   Range: HH-MM-SS, with HH between 00 and 23, MM between 00 and 59, SS between 00 and 59.
   Default: 00-00-00 after start-up with default DB current NE value, if not supplied in the command.

Output format

If the ED-DAT request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The following attributes of the system entity is updated due to that command

1. *date* as specified in the command input
2. *time* as specified in the command input.

If either *date* or *time* is not supplied in the command parameter, this value is left unchanged.

The update is reported by means of an ED-DAT notification for the system.

Update affected entities

If ED-DAT is changed so that a performance monitoring period is crossed, than one history bin will be lost. In every case the *validity* parameter is modified accordingly (see “RTRV-PM: Retrieve performance monitoring” (p. 3-2731)).

Example input/output

```
ED-DAT:LT-UNITE:789::123456::01-12-31,11-50-30;

   LT-UNITE:789 01-12-31 11:51:41
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-DAT`** command.

Validation rule: Incorrect date/time format

If the NE receives an ED-DAT command with an incorrectly formatted date and/or time, the following error response is returned. This check includes checks for invalid days within a month (e.g. no 30th February).

Error code: `IDNV – Input, Data Not Valid, incorrect date/time format`

Related TL1 messages

"RTRV-HDR: Retrieve header" (p. 3-2417)

References

GR-199-CORE

ED-DAT notification

Output format

Syntax: **ED-DAT**:::[*date*][,*time*]

Output parameters

Command parameters

1. **date** – Date
   The *date* specifies the current date as YY-MM-DD. If the parameter *time* is not specified, the current value is not changed, but then *date* must be specified for the **ED-DAT** command to complete successfully.
   Type: value set.
   Range: YY-MM-DD, with YY between 00 and 99, MM between 01 and 12, DD between 01 and 31.
   Default: 00-01-01 after start-up with default DB, current NE value, if not supplied in the command.
2. **time** – Time
   The *time* specifies the current time of day as HH-MM-SS. If the parameter *date* is not specified, the current value is not changed, but then *time* must be specified for the **ED-DAT** command to complete successfully.
   Type: value set.
   Range: HH-MM-SS, with HH between 00 and 23, MM between 00 and 59, SS between 00 and 59.
   Default: 00-00-00 after start-up with default DB, current NE value, if not supplied in the command.

Detailed behavior description

Date and time in the ED-DAT notification

The *date* and *time* parameters in the header of the **REPT DBCHG** notification as well as the **REPT DBCHG** output parameters *chgdat* and *chgtm* indicate the NE date and time when the **ED-DAT** command has been invoked, that means the date and time before the change.

The **ED-DAT** spec block parameters *date* and *time* contained in the **REPT DBCHG** command show the new NE date and time after the change.

Example output

Example of an **ED-DAT** command invoked on November 14 2007 at 17:50:10:

```
ED-DAT:UNITE-NE:789::123456::07-11-16,12-20-00;

   UNITE-NE:789 07-11-14 17:50:10
M  123456 COMPLD
```

```
;

   UNITE-NE  07-11-14  17:50:10
A  078  REPT  DBCHG
   ”Time=17-50-10,DATE=07-11-14:ED-DAT:::date=07-11-16,time=12-20-00”
;
```

ED-EC1: Edit EC1 port

The command **ED-EC1** is used to retrieve all parameters related to the electrical EC1 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax:

ED-EC1:*tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Access Identifier of a EC1 port.
   Type: AID
   Range: SPortAID of EC1 port. The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **eif** – Electrical Interface Type
   This parameter specifies the type of input to which the port is provisioned.
   Type: Alphanumeric value set.
   Range: DS3, EC1 (default)
   The default vlaue EC1 leaves the interface type unchanged, if the parameter is omitted.
2. **felpm** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section/line.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)
3. **lbo** – Line Build Out
   This parameter defines the cable length applicable for the EC1 port.
   Type: Alphanumeric value set.

Range:

- SHORT (default): 0 … 68.5 m (0 … 225 ft)
- LONG: 68.5 … 137 m (225 … 450 ft).

4. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range: ENABLE: set a forced DUS/DNU; DISABLE (default): clear a forced DUS/DNU
5. **`nelpm`** – Near-End Line PM Enable
   Enable or disable near-end multiplex section (near-end line) performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
6. **`pmmsfeses`** – Performance Monitoring Line Far-End SES Threshold
   This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
   Type: numeric value set.
   Range: 1 – 6400.
   Default: 51
7. **`pmmsneses`** – Performance Monitoring Line Near-End SES Threshold
   This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
   Type: numeric value set
   Range: 1 – 6400.
   Default: 51
8. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
9. **`pmrsneses`** – Performance Monitoring Section Near-End SES Threshold
   This parameter describes the Section Near-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
   Type: numeric value set.
   Range: 1 … 64000
   Default: 52

10. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when sdsfmode=POISSON.
    Type: numeric value set.
    Range: (as exponents of 10): -5 -6 (default), -7, -8, -9
11. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
12. **`sfthr`** – Signal Failure Threshold
    Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when sdsfmode=POISSON
    Type: numeric value set
    Range: (as exponents of 10): -3 (default), -4, -5
13. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
14. **`sonet`** – SONET ASAP
    SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports of corresponding line rate (see RTRV-STM4, output parameter sdhpn) to be compatible with current generation applications.
    Type: Alphanumeric string
    Range: String with a maximum of 24 characters. A hyphen (“-”) is allowed in the name additionally.
    Default: DEFAULT.
15. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
16. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted.
    Type: Hexadecimal digits
    Range: 1 byte string, One byte is expressed as two hexadecimal digits (example value: 6D).
    Default: 01Hex

Output format

If the ED-EC1 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **RTRV-HDR** command.

Detailed behavior description

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command. The modification is reported by means of an ED-EC1 notification for port (except for parameter eif; see below).

Switching of interface type - Ignore other parameters

If additional parameters beside eif are specified with the ED-EC1 command they will be ignored by the NE.

Switching of interface type - Impact on port and Tributary Entities

Setting *eif* to T3 shall

- delete the STS-1 path termination entity related to the EC1 port,
- delete the EC1 port entity,
- create one T3 port entity with the defined default values for the port parameters,
- create one STS-1 path termination with the default values for the PTF parameters.

The deletion and creation of ports and tributaries are notified via the appropriate DLT and ENT notifications.

Switching of interface type - Impact on Fault Management

Setting *eif* to T3 shall

- clear all active alarms for the EC1 port and the related tributary
- re-evaluate the alarm status after the creation of the T3 port and the path termination
- raise all active alarms for the T3 port and the tributary.

The clearing and raising of alarms are notified with the appropriate notifications.

Switching of interface type - Impact on Performance Monitoring

Setting *eif* to T3 shall

- disable all active PM monitoring points for the EC1 port and the related tributary
- clear all current and history bins for the EC1 port and the related tributary.

Example input/output

```
ED-EC1:UNITE-NE:1-1-#-#-1-36:123456:::FELPM=ENABLE,NELPM=ENABLE;

   UNITE-NE 04-12-23 08:00:00
M  123456 COMPLD
;

   UNITE-NE 04-12-23 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=04-12-23:ED-EC1:1-1-#-#-1-1::felpm=enable,nelpm=enable:"
;
```

Error responses

See definition in "RTRV-HDR: Retrieve header" (p. 3-2417) command error responses section for general validation rules. The error responses listed there also apply to the ED-EC1 command.

Validation rule: Check CCB against *ptype* and *eif*

Check wether the CCB fits to *ptype* and *eif*.

Error code:`IENE – Input, Entity Not Exists`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code:`IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: no cross-connections for interface type change

Check that eif is not changed when a facility loopback is still on this port.

Error code:
`SNVS – Status, Not in Valid State, cross-connections still exist`

Validation rule: no conversion cross-connections for interface type change

Check that *eif* is not changed when conversion cross-connections are still on this port.

Error code:
`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: no XC loopbacks for interface type change

Check that *eif* is not changed when cross-connections loopback is still on this port.

Error code:

`SNVS – Status, Not in Valid State, a cross-connect loopback exists`

Validation rule: No facility loopbacks for interface type change

Check that *eif* is not changed when a facility loopback is still on this port.

Error code:

`SNVS – Status, Not in Valid State, facility loopbacks still exist`

Validation rule: no test access session for interface type change

Check that *eif* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Check whether system is in maintenance condition

Check that *eif* is not changed when the system in is maintenance condition.

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loobackp is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (strcout) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data Not Consistent, strcout length must match the given outgoing J0 trace mode`

Related TL1 messages

- “RTRV-EC1: Retrieve EC1 port” (p. 3-2332)
- “ED-EC1 notification” (p. 3-325)

- “ENT-EC1 notification” (p. 3-1515)
- “DLT-EC1 notification” (p. 3-163)

References

GR-199-CORE (ED-*rr*)

ED-EC1 autonomous notifications

LOS for transmission port EC1 clears

If LOS is cleared and current port mode is AUTO, then portmode is updated (new value: MON). This update is reported via ED-EC1 notification with parameter pmode.

ED-EC1 notification

Output format

Command syntax:

ED-EC1:*tid:aid:ctag:::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: EC1 port AID

Spec block parameters

1. **felpm** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section/line.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)
2. **frcdus** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range: ENABLE: set a forced DUS/DNU; DISABLE (default): clear a forced DUS/DNU
3. **fsflpbkstat** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output aid. A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback exististing for the aid.
   Note: not both loopbacks (lpbk and fsflpbkstat) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range: YES, NO (default).
4. **lbo** – Line Build Out
   This parameter defines the cable length applicable for the EC1 port.
   Type: Alphanumeric value set.
   Range:
   - SHORT (default): 0 … 68.5 m (0 … 225 ft)
   - LONG: 68.5 … 137 m (225 … 450 ft).

5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output aid. A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback exististing for the port.
   Type: Alphanumeric value set.
   Range: YES, NO (default)
6. **`nelpm`** – Near-End Line PM Enable
   Enable or disable near-end multiplex section (near-end line) performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)
7. **`pmmsfeses`** – Performance Monitoring Line Far-End SES Threshold
   This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
   Type: numeric value set.
   Range: 1 – 6400.
   Default: 51
8. **`pmmsneses`** – Performance Monitoring Line Near-End SES Threshold
   This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
   Type: numeric value set
   Range: 1 – 6400.
   Default: 51
9. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
10. **`pmrsneses`** – Performance Monitoring Section Near-End SES Threshold
    This parameter describes the Section Near-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
    Type: numeric value set.
    Range: 1 … 64000
    Default: 52

11. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when sdsfmode=POISSON.
    Type: numeric value set.
    Range: (as exponents of 10): -5 -6 (default), -7, -8, -9
12. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
13. **`sfthr`** – Signal Failure Threshold
    Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when sdsfmode=POISSON
    Type: numeric value set
    Range: (as exponents of 10): -3 (default), -4, -5
14. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
15. **`sonet`** – SONET ASAP
    SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports of corresponding line rate (see RTRV-STM4, output parameter sdhpn) to be compatible with current generation applications.
    Type: Alphanumeric string
    Range: String with a maximum of 24 characters. A hyphen (“-”) is allowed in the name additionally.
    Default: DEFAULT.
16. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
17. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted.
    Type: Hexadecimal digits
    Range: 1 byte string, One byte is expressed as two hexadecimal digits (example value: 6D).
    Default: 01Hex

Example Output

```
ED-EC1:UNITE-NE:1-1-#-#-21-1:123456:::FELPM=ENABLE,NELPM=ENABLE;

   UNITE-NE 04-12-23 08:00:00
M  123456 COMPLD
;

   UNITE-NE 04-12-23 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=04-12-23:ED-EC1:1-1-#-#-21-1::felpm=enable,nepm=enable:”
;
```

ED-EPORT: Edit Ethernet port

The **ED-EPORT** command modifies the properties of an Ethernet port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **ED-EPORT**:*modifier*:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **modifier**
   This modifier specifies the type of the Ethernet port at the LAN side.
   Type: Alphanumeric value set.
   Range: 1GE representing a 1 GE port on a GE1 pack , 1GEPL representing a 1 GE port on a GE10PL1 pack, 10GEPL representing a 10 GE port on a GE10PL1 pack.
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   This parameter specifies the Ethernet port to be operated on.
   Type: AID
   Range: EPortAID as defined in AID structure. The usage of the following values are excluded:
   - *all* in the BayAID definition
   - *all* in the Simple_ShelfAID definition
   - *usall* in the Simple_UnivSlotAID definition
   - *all* in the Simple_UnivSlotAID definition
   - *all* in the Simple_EPortAID definition.
4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameter

1. **`autoneg`** – Auto Negotiation Mode
   This parameter controls auto-negotiation process to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE or 10GEPL. If the parameter *autoneg* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default); DISABLE.
2. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer LAN port or a network LAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER (default), for a customer LAN port; NETWORK, for a network LAN port.
3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (i.e. from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - 1, 2, 3, … 4092, 4093
   - UNAVAILABLE (default).

   To express the state UNAVAILABLE system internally the invalid VLAN ID 4094 is used.

5. **`droppause`** – Drop Incoming Pause Frames
   This parameter controls whether incoming pause frames are dropped (value = ENABLE) or not (value = DISABLE). This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter droppause is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
   Default: DEFAULT.
6. **`enetpn`** – Ethernet Alarm Severity Assignment Profile
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
7. **`enettca`** – Ethernet TCA Profile
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for Ethernet ports.
   Type: Alphanumeric string.
   Range: 1 … 24 characters length.
   Default: DEFAULT0.
8. **`epmode`** – Ethernet Port Monitoring Mode
   This parameter specifies the monitoring mode of the Ethernet port. Alarms for this port will be reported if in state MON otherwise not. The configuration of epmode will change from AUTO to MON if a valid signal is applied to that port.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default), for automatic
   - MON, for monitored
   - NMON, for not monitored.
9. **`eppm`** – Ethernet Port Performance Monitoring
   This parameter specifies whether the performance monitoring is enabled or disabled for this port.
   Type: Alphanumeric value set.
   Range:
   - DISABLE (default)
   - ENABLE
10. **`fcmd`** – Flow Control Mode
    This parameter specifies the configuration for the pause operation of this port. If the input parameter modifier = 1GE, then the parameter fcmd is supported only in case that the bridge operates in backward-compatible mode (vlantagmd = BCMD).
    Type: Alphanumeric value set.

Range:

- AUTO, for auto-negotiate
- DISABLE, for pause mode bidirectionally disabled
- ASYMTOPARTNER, for pause mode having receive disabled, transmit enabled
- ASYMTOLOCAL, for pause mode having transmit disabled, receive enabled
- SYM, for pause mode bidirectionally enabled.

Restrictions: If the parameter *fcmd* is not supported, then the parameter is omitted.
The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GE.
The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = ENABLE. The value AUTO is notsupported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = DISABLE.
The value AUTO is not supported, if the input parameter *modifier* = 10GEPL.
Possible combinations of the parameters *modifier*, *autoneg*, *vlantagmd* (parameter of the referring circuit pack entity, see "RTRV-EQPT (CP): Retrieve equipment (circuit pack)" (p. 3-2375) for definition) and *fcmd* are as follows:

Input parameter *modifier*	**In-/output parameter *autoneg***	**In-/output parameter *vlantagmd***	**In-/output parameter *fcmd***
1GE	not supported	BCMD	AUTO, DISABLE
		other than BCMD	not supported
1GEPL	ENABLE	not supported	AUTO, DISABLE
	DISABLE		DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM
10GEPL	not supported	not supported	DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM

Default:

- AUTO (for *modifier* 1GE or 1GEPL)
- SYM (for *modifier* 10GEPL).

11. **`rcf_htmr`** – Remote Client Fail Hold Off Timer
The parameter specifies the hold-off time for applying the LPT consequent action to disable the laser at the LAN port. The hold-off timer is only applicable if there is a strict one-to-one relationship between the LAN port in question and its related WAN port.
This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_htmr* is not supported, then the parameter is omitted.
Type: numeric string [ms].
Range: 0 … 2000 ms in 100 ms steps
Default: 0.

12. **`rcf_linkdown`** – Link Pass Through Mode
    This parameter controls the consequent action (to switch-off the laser upon receipt of a client signal fail [CSF] indication) to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_linkdown* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE.
    Default: ENABLE.
13. **`rptmode`** – Repeater Mode
    This parameter sets the bridge device for this LAN port into the repeater mode (promiscuous mode), valid for a point to point connection between one LAN and one WAN port. Operating in repeater mode imposes some restrictions. See detailed behavior section.
    If the input parameter *modifier* = 1GE, then the parameter rptmode is meaningful only if in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
14. **`servcond`** – Service Condition
    This parameter specifies the service condition for this port. Some configuration possibilities like loopbacks may be dependent on the state of *servcond*.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default)
    - OUTOFSERVICE.
15. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. If the input parameter modifier = 1GE, then this parameter is meaningful only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter stdis is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.

Range:

- DISABLE, STP is disabled, i.e. the port is not used in the spanning tree protocol
- ENABLE (default), STP is enabled, i.e. the port is used in the spanning tree protocol.

16. **`stpcost`** – Spanning Tree Path Cost
The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
If the parameter *stpcost* is not supported, then the parameter is omitted.
Type: Integer.
Range: 1 to 200000000.
Default: 20000.
The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

17. **`stpp`** – Spanning Tree Port Priority
This parameter specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
Type: Integer.
Range: 0 to 255.
The stepping as recommended by the standard (steps of 16 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 240″ is not done by the NE.
Default: 128.

18. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding. Two thresholding methods are applicable: TR-only. (also known as the implicit reset method), TR-RTR. (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR (default).

19. **vvidls** – Valid VLAN Identifier List (static)
    This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using '&' and '&&' . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (cnprole = NETWORK).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
    Type: Alphanumeric string and the character '&'.
    Range:
    - Value of each single member (valid VLAN identifier) in the list 1, 2, ..., 4092, 4093
    - In case the list has no members (empty list) -1.

    The length of the list is limited only by the usual system limitations like e.g. TL1 line length. The number of entries in this list is limited to 64 by some NE-internal restrictions. Ranging and grouping by using the operators '&' and '&&' has no impact on the number of entries, it only reduces the length of the list being transported to/from CIT/EMS.
    Default: -1.
    Example 1:
    vvidls=-1. The list has no members (empty list).
    Example 2:
    vvidls=3&5&40&&45. This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.
    Example 3:
    vvidls=1&&4093 This list has the members 1,2, 3, ..., 4092, 4093.

Output format

If the NE fully complies with the ED-EPORT request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The ED-EPORT generates a REPT DBCHG message.

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command. In case of defined options for a specific parameter, the related parameter can only be modified if all conditions to support this parameter are given.

The modification is reported by means of an ED-EPORT notification.

Update affected entities due to a change of the repeater mode

A change of the value of the parameter *rptmode* shall have the following side effects: Any change of the value from DISABLE to ENABLE or vice versa shall reset the parameter dftvlan and vvidls for this port to their default values.

During repeater mode operation no VLAN ingress filtering for tagged frames or classification for untagged frames is done. It is not intended to keep the VLAN settings being active before a switch to the repeater mode for later restoration. This requirement is only applicable if the input parameter *modifier* = 1GE.

Relation of repeater mode - *rptmode* and VLAN tagging mode - *vlantagmd* of related slot

When *vlantagmd* is set to BCMD (backward compatible mode) the value of *rptmode* is relevant. When *vlantagmd* has a value different from BCMD the value of *rptmode* is irrelevant and changes of *rptmode* is not allowed. When switching *vlantagmd* back to BCMD the value of *rptmode* from the database becomes relevant again. This requirement is only applicable if the input parameter *modifier* = 1GE.

Reset to default values when switching from VLAN-aware p2p mode to repeater mode

When vlantagmd is set to BCMD and rptmode is changed from DISABLE to ENABLE the following parameters of the related EPORTs are set to their default values:

- *dftpri*, *dftvlan*, *vvidls*.
  This requirement is only applicable if the input parameter *modifier* = 1GE.

Implicit changes of *fcmd* while configuring *autoneg*

If the parameter *autoneg* is changed from the value DISABLE to ENABLE without modifying the parameter fcmd within the same command then the parameter fcmd will change implicitly to the value AUTO unless it had the value DISABLE previously.

If the parameter *autoneg* is changed from the value ENABLE to DISABLE without modifying the parameter fcmd within the same command then the parameter fcmd will change implicitly to the value DISABLE provided it had a different value previously.

Any implicit change have to be reported within the same ED-EPORT notification. This requirement is only applicable if the input parameter *modifier* = 1GEPL or if the input parameter *modifier* = 10GEPL.

Implicit changes of anegst while configuring autoneg

If the parameter *autoneg* is changed from the value DISABLE to ENABLE then the parameter anegst will change implicitly to a different value than DISABLE.

If the parameter *autoneg* is changed from the value ENABLE to DISABLE then the parameter anegst will change implicitly to the value DISABLE provided the value has not yet been previously indicated.

Any implicit change will not be reported but will be visible within the response to a subsequent RTRV-EPORT command being issued for the related port. This requirement is only applicable if the input parameter *modifier* = 1GEPL or if the input parameter *modifier* = 10GEPL.

Although usually no requirements are given for a parameter which changes during the modification of another parameter and which is not subsequently notified, here is an exception to indicate that the indicated value range of the parameter (*anegst*) will change depending on the configuration value of the other parameter (*autoneg*).

Implicit enabling/disabling of GVRP when enabling/disabling STP for a port

When enabling/disabling STP for a network port via parameter *stdis* GVRP is implicitly enabled/disabled for this port (if enabled via parameter *gvrpmode* of the virtual switch).

This requirement is only applicable if the input parameter *modifier* = 1GE.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. Stop PM counting
2. All current bins related to the PM monitor is deleted
3. All history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of ’pm enable attributes’. PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Example input/output

```
ED-EPORT-1GE:LT-UNITE:1-1-#-#-1-1:123457:::epmode=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123457 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-EPORT** command.

Validation rule: *modifier* consistent with given port-rate

The system shall verify that the modifier parameter is consistent with the rate of the port (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-EPORT** command for applicable combinations.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with existing port rate`

Validation rule: *modifier* consistent with the related circuit pack type

The system shall verify that the modifier parameter is consistent with the related pack type (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-EPORT** command for applicable combinations.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with the related circuit pack type`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: servcond must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: No Modification of fcmd if *vlantagmd* = 802.1TAG

No modification of the flow control mode fcmd is allowed if the VLAN tagging mode vlantagmd of the pack, to which the specified EPORT belongs to, is set to 802.1TAG.

Error code:

`SNVS – Status, Not in Valid State, Modification of flow control mode not allowed`

Validation rule: Default VLAN ID not allowed in repeater mode

No VLAN ingress classification with dftvlan is possible during repeater mode operation. The default VLAN ID has to be the value ″unavailable″ during this state.

Error code:

`SNVS – Status, Not in Valid State, Active Repeater Mode for this Port does not allow dftvlan to be modified`

Validation rule: VLAN filtering database not allowed in repeater mode

No VLAN filtering database, extracted from the vvidls, is used during repeater mode operation. The valid VLAN list has to be an empty list during this state.

Error code:

`SNVS – Status, Not in Valid State, Active Repeater Mode for this Port does not allow vvidls to be modified`

Validation rule: Parameter modification not allowed in backward compatible mode

No modification of the following parameters is allowed if the VLAN tagging mode *vlantagmd* of the pack, to which the specified EPORT belongs to, is set to backward compatible mode (BCMD): *cnprole*, *stdis*, *stpp*, *stpcost*, *rstpmd*.

Error code:

`SNVS – Status, Not in Valid State, Modification not allowed in Backward Compatible Mode`

Validation rule: No modification of *rptmode* if *vlantagmd* = 802.1TAG

No modification of the repeater mode *rptmode* is allowed if the VLAN tagging mode *vlantagmd* of the pack, to which the specified EPORT belongs to, is set to 802.1TAG.

Error code: `SNVS – Status, Not in Valid State, Modification not allowed`

The rationale for this restriction is to avoid side-effects when changing *rptmode*.

Validation rule: Parameters not supported on GE1 pack

The command is denied for ports on the GE1 pack, if one of the parameters *rcf_htmr* or *rcf_linkdown* is specified in the command.

Error code:

`IDNV – Input, Data Not Valid, Parameter(s) present which are not allowed`

Validation rule: Default VLAN ID is already assigned for this system

The default VLAN ID of a LAN port cannot have the same value as the default VLAN Id of another LAN port in the system or any other instantiated VLAN in the system, except for the value 4094.

Error code:

`IDNC-Input, Data, Not Consistent, Default VLAN ID is already assigned for this System`

System is here to be seen with the granularity of a GE1 circuit pack.

Validation rule: BCMD mode - Default VLAN ID is no allowed member of valid VLAN list for this System

If the system is running in backward-compatible mode (*vlantagmd* = BCMD), the default VLAN ID of a LAN port cannot be included into the valid VLAN list of any LAN port within the system.

Error code:

`IDNC-Input, Data Not Consistent, Default VLAN ID is not an allowed member of valid VLAN list`

System is here to be seen with the granularity of a GE1 circuit pack.

Validation rule: Maximum number of entries in valid VLAN ID list exceeded

If the number of VLAN IDs specified with parameter *vvidls* is greater than 247 the following error code is returned. For calculating the number of VLAN IDs being specified the grouping and ranging operators ’&’ and ’&&’ have to be resolved first.

Error code:

`SSRE – Status, System Resources Exceeded, limit of VLAN Identifiers exceeded`

Validation rule: GE10PL1 Pack specific validation rule: No fix flow control settings if autonegotiation enabled

The system shall reject an attempt to modify the parameter fcmd to the values ASYMTOPARTNER, ASYMTOLOCAL and SYM if the parameter *autoneg* is not modified to the value DISABLE within the same command and the current setting of the parameter *autoneg* = ENABLE.

Error code:

`IDNC – Input, Data Not Consistent, Flow Control settings do not match to existing autonegotiation mode`

Validation rule: GE10PL1 Pack specific validation rule: No auto flow control settings if autonegotiation disabled

The system shall reject an attempt to modify the parameter fcmd to the value AUTO if the parameter *autoneg* is not modified to the value ENABLE within the same command and the current setting of the parameter *autoneg* = DISABLE.

Error code:

`IDNC – Input, Data Not Consistent, Flow Control settings do not match to existing autonegotiation mode`

Related TL1 message

- “RTRV-EPORT: Retrieve Ethernet port” (p. 3-2358).

ED-EPORT notification

Output format

The **ED-EPORT** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **ED-EPORT** notification has the following specific syntax:

ED-EPORT-*modifier:aid::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the Ethernet port to be operated on.
   Type: AID
   Range: EPortAID as defined in AID structure. The usage of the following values are excluded:
   - *all* in the BayAID definition
   - *all* in the Simple_ShelfAID definition
   - *usall* in the Simple_UnivSlotAID definition
   - # in the Simple_UnivSlotAID definition
   - all in the Simple_EPortAID definition.
2. **modifier**
   This modifier specifies the type of the Ethernet port at the LAN side.
   Type: Alphanumeric value set.
   Range: 1GE representing a 1 GE port on a GE1 pack , 1GEPL representing a 1 GE port on a GE10PL1 pack, 10GEPL representing a 10 GE port on a GE10PL1 pack.

Spec block parameters

1. **autoneg** – Auto Negotiation Mode
   This parameter controls auto-negotiation process to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE or 10GEPL. If the parameter *autoneg* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default); DISABLE.
2. **cnprole** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer LAN port or a network LAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- USTOMER for a customer LAN port
- NETWORK for a network LAN port

Default: CUSTOMER

3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: HIGH, LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to ingressing frames (tagging) . This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: 1, 2, 3, … 4092, 4093; UNAVAILABLE (default)
5. **`droppause`** – Drop Incoming Pause Frames
   This parameter controls whether incoming pause frames are dropped (value = ENABLE) or not (value = DISABLE). This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter droppause is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
   Default: DEFAULT.
6. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Type: Alphanumeric string A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length
   Default: DEFAULT
7. **`enettca`** – Ethernet TCA Profile Name
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for Ethernet ports.
   Type: Alphanumeric string
   Range: 1 … 24 characters length
   Default: DEFAULT0

8. **`epmode`** – Ethernet Port Monitoring Mode
   This parameter specifies the monitoring mode of the Ethernet port.
   Type alphanumeric value set
   Range:
   - AUTO for automatic (default)
   - MON for monitored
   - NMON for not monitored.
9. **`eppm`** – Ethernet Port Performance Monitoring
   This parameter specifies whether the performance monitoring is enabled or disabled for this port.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
10. **`fcmd`** – Flow Control Mode
    This parameter specifies the configuration for the pause operation of this port. If the input parameter modifier = 1GE, then the parameter fcmd is supported only in case that the bridge operates in backward-compatible mode (vlantagmd = BCMD).
    Type: Alphanumeric value set.
    Range:
    - AUTO, for auto-negotiate
    - DISABLE, for pause mode bidirectionally disabled
    - ASYMTOPARTNER, for pause mode having receive disabled, transmit enabled
    - ASYMTOLOCAL, for pause mode having transmit disabled, receive enabled
    - SYM, for pause mode bidirectionally enabled.

    Restrictions: If the parameter *fcmd* is not supported, then the parameter is omitted.
    The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GE.
    The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = ENABLE. The value AUTO is notsupported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = DISABLE.
    The value AUTO is not supported, if the input parameter *modifier* = 10GEPL.

    Possible combinations of the parameters modifier, autoneg, vlantagmd (parameter of the referring circuit pack entity, see RTRV-EQPT for definition) and fcmd are as follows:
    - Input parameter *modifier*=1GE, In-/Output parameter *autoneg* = not supported, In-/Output parameter *vlantagmd* = other than BCMD, In-/Output parameter *fcmd* = AUTO, DISABLE
    - Input parameter *modifier*=1GE, In-/Output parameter *autoneg* = not supported, In-/Output parameter *vlantagmd* = BCMD, In-/Output parameter *fcmd* = not supported

- Input parameter *modifier*=1GEPL, In-/Output parameter *autoneg* = ENABLE, In-/Output parameter *vlantagmd* = not supported, In-/Output parameter *fcmd* = AUTO, DISABLE
- Input parameter *modifier*=1GEPL, In-/Output parameter *autoneg* = DISABLE, In-/Output parameter *fcmd* = DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM

Default:

- AUTO (for *modifier* 1GE or 1GEPL)
- SYM (for *modifier* 10GEPL).

11. **`fsflpbkstat`** – Far Side Facility Loopback Status
This parameter reports the configuration of the far-side facility loopback of this port. A far-side loopback means that the signal on the outgoing port is looped to the incoming port (Tx -> Rx). For GE1 there is only this type of facility loopback. For changing the loopback operation refer to OPR-LPBK of section Transmission and Cross-Connection Handling, Port
Type: Alphanumeric value set.
Range:
    - YES, for enabled loopback
    - NO (default), for disabled loopback.
12. **`rcf_htmr`** – Remote Client Fail Hold Off Timer
The parameter specifies the hold-off time for applying the LPT consequent action to disable the laser at the LAN port. The hold-off timer is only applicable if there is a strict one-to-one relationship between the LAN port in question and its related WAN port.
This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_htmr* is not supported, then the parameter is omitted.
Type: numeric string [ms].
Range: 0 … 2000 ms in 100 ms steps
Default: 0.
13. **`rcf_linkdown`** – Link Pass Through Mode
This parameter controls the consequent action (to switch-off the laser upon receipt of a client signal fail [CSF] indication) to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_linkdown* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE.
Default: ENABLE.
14. **`rptmode`** – Repeater Mode
This parameter sets the bridge device for this LAN port into the repeater mode (promiscuous mode), valid for a point to point connection between one LAN and one WAN port. Operating in repeater mode imposes some restrictions. See detailed behavior section.
If the input parameter *modifier* = 1GE, then the parameter rptmode is meaningful

only if in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

15. **`servcond`** – Service Condition
This parameter specifies the service condition for this port.
Type alphanumeric value set
Range: INSERVICE (default), OUTOFSERVICE

16. **`stdis`** – Spanning Tree Protocol Disable
This parameter allows to enable or disable the Spanning Tree Protocol for the given port. If the input parameter modifier = 1GE, then this parameter is meaningful only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter stdis is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

    Range:

    - DISABLE, STP is disabled, i.e. the port is not used in the spanning tree protocol
    - ENABLE (default), STP is enabled, i.e. the port is used in the spanning tree protocol.

17. **`stpcost`** – Spanning Tree Path Cost
The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
If the parameter *stpcost* is not supported, then the parameter is omitted.
Type: Integer.

Range: 1 to 200000000.
Default: 20000.
The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

18. **stpp** – Spanning Tree Port Priority
This parameter specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
Type: Integer.
Range: 0 to 255.
The stepping as recommended by the standard (steps of 16 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 240″ is not done by the NE.
Default: 128.

19. **tcamd** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
   - TR-only. (also known as the implicit reset method)
   - TR-RTR. (also known as the explicit reset method)

   Type: Alphanumeric value set.
Range: TR, TRRTR (default)

20. **vvidls** – Valid VLAN Identifier List
This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&) acc. GR-833-CORE Input Commands and Responses.
The list is packed by the NE and the management systems as good as possible by using ’&’ and ’&&’ .
The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list vvidlsd. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (cnprole = NETWORK).
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *vvidls* is not supported, then the parameter is omitted.
Type: Alphanumeric string and the character ’&’.

Range: Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093. In case the list has no members (empty list) -1 (default).
The length of the list is limited by the usual system limitations like e.g. TL1 line length and some NE internal restrictions as described below. The number of entries in this list is limited to 247 by some NE-internal restrictions. Ranging and grouping by using the operators '&' and '&&' has no impact on the number of entries, it only reduces the length of the list being transported to/from CIT/EMS.
Example 1: *vvidls=-1*. The list has no members (empty list).
Example 2: v*vidls=3&5&40&&45*. This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.
Example 3: *vvidls=1&&4093*. This list has the members 1,2, 3, …, 4092, 4093.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-EPORT-1GE:1-1-#-#-1-1::fcmd=DISABLE″
;
```

ED-EQPT (CP): Edit equipment (circuit pack)

The **ED-EQPT** command modifies the provisioning values associated with equipment. Circuit packs must first be provisioned with the **ENT-EQPT** command or via auto provisioning before they can be modified.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-EQPT***:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of an equipment.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for for circuit pack in any slot
   - LoxcAID ::= UnivSlotAID-loxc for LOXCpack in universal slots.

   The keyword **usall** is not allowed for Simple_UnivSlotAID.
   The keyword **xcall** is not allowed for Simple_XCSlotAID.
   The keyword **ctlall** is not allowed for Simple_CTLSlotAID.
   The keyword **all** is not allowed for Simple_SlotAID.
   PowerAID for power interface: The keyword **piall** is not allowed for PowerAID.
   EciAID for ECI interface The keyword **eciall** is not allowed for EciAID.
   TimingInterfaceAID for TI Interface board The keyword ″tiall″ is not allowed for TimingInterfaceAID.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`dcn_ipad`** – DCN IP address for CLI/SNMP
   DCN IP address for CLI/SNMP management of the card.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255.
   The values 0.0.0.0, 1.1.1.1, 127.0.0.1, and 224.0.0.0 to 255.255.255.255 are excluded.
   Default: Empty string.
2. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the circuit pack. This parameter is reported for a CpAID , PowerAID, EciAID, TimingInterfaceAID and LoxcAID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
3. **`eqptprotpn`** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT for the equipment.
   Option: This parameter is only reported for a CpAID, LoxcAID or PowerAID.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
   Default: DEFAULT
4. **`es64ucap`** – ES64U capacity
   Switch capacity of the ES64U pack.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0 – ES64U in suspended state, initial value after ES64U CP creation.
   - 20 – ES64U active or member of equipment protection group, capacity 20 Gbit/s.

   Default: 0.
5. **`vlantagmd`** – VLAN Tagging Mode
   VLAN tagging mode of the circuit pack. This parameter is relevant only for circuit packs of type GE1 (that means, *nvmname* = GE1).
   Type: value set.
   Range:
   - BCMD (default), for the backward compatible mode
   - 802.1tag, for the VLAN tag based mode according to IEEE802.1Q.

   Default: BCMD

6. **`loswcap`** – LOXC switch capacity
   Switch capacity of the LOXC pack. This parameter is relevant only for circuit packs of type LOXC, thus the parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   If *nvmname*==LOXC:
   - 0 (default): LOXC in suspended state, initial value after LOXC CP creation
   - 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s

   If *nvmname* = LOXC40G2S or LOXC40G3S:
   - 0: LOXC in suspended state, initial value after LOXC CP creation
   - 5: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s
   - 40: LOXC active or member of equipment protection group, switch capacity 40 Gbit/s
7. **`tmuxcap`** – TransMUX capacity
   Switch capacity of the TransMUX pack.
   This parameter is relevant only for circuit packs of type TransMUX (that means, *nvmname* = TransMUX), thus the parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0: TransMUX in suspended state, initial value after TransMUX CP creation.
   - 2G5: TransMUX active or member of equipment protection group, capacity 2.5 Gbit/s

   Default: 0
8. **`istd`** – Interface Standard of the Circuit Pack
   Interface standard supported by the circuit pack. This parameter is supported on GE10PL1 and ES64U circuit packs. On the circuit packs, the parameter *istd* defines the interface standard of the virtual concatenation groups.
   Option: If the parameter *istd* is not supported, then the parameter is omitted.
   Type: Value set.
   Range:
   - for GE10PL1: SDH or SONET
   - for ES64U: SONET or UNDEFINED (ES64U not used, *es64ucap* = 0)

   Default:
   - for GE10PL1: The default is derived from the system parameter *istddflt*.
   - for ES64U: UNDEFINED.

Output format

If the **ED-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command modifies the value of *eqptpn*, *eqptprotpn* for a circuit pack entity as indicated by the *spec_block* parameter.

The update is reported by means of an **ED-EQPT** notification for circuit pack.

Change of VLAN tagging mode: changing *vlantagmd* to 802.1TAG - update affected entities

If *nvmname* is GE1 a change of parameter vlantagmd from BCMD to 802.1TAG has the following side effects:

- The command shall create one instance of entity virtual switch. This are notified by a ENT-VRTSW notification
- The attribute *fcmd* of all related EPORTs are set to DISABLED.
- The attribute *rptmode* of all related EPORTs are set to DISABLED.
- All other attributes of related EPORTs and VCGs shall remain unchanged.

Change of VLAN tagging mode: changing vlantagmd to BCMD - update affected entities

If *nvmname* is GE1 a change of parameter vlantagmd from 802.1TAG to BCMD has the following side effects: The command shall delete the instance of entity virtual switch. This are notified by a DLT-VRTSW notification.

The following attributes of the related EPORTs are set to their default values:

- *fcmd*, *stdis*, *stpp*, *stpcost*.

The following attributes of the related VCGs are set to their default values:

- *cnprole*, *stdis*, *stpp*, *stpcost*.

All other attributes of related EPORTs and VCGs shall remain unchanged.

Change of LOXC switch capacity: changing of *loswcap*

The paramter *loswcap* can be set with the ED-EQPT(cp) command. Changing this parameter to a different value than 0 is only allowed if there is no active LOXC pack in the system.

The term active LOXC pack refers to:

- either an unprotected pack with loswcap unequal to 0,
- or an active pack in an LOXC equipment protection group.

Standby LOXC packs are not considered as active LOXC pack.

Change of LOXC switch capacity: update of LO tributary Standard

If the parameter *loswcap* is set to a different value than 0, then the parameter *lostd* is set to the current value of the parameter *istddflt*. If the parameter *loswcap* is set to 0, then the parameter *lostd* is set to UNDEFINED.

Change of Interface Standard - behavior due to interface standard change on GE10PL1 pack

A change of *istd* is not dependent from pending alarms or active performance monitoring for the individual tributaries. A change of *istd* changes the interface standard of all existing VCG ports on the pack.

VCG ports are set to default values:

- *enettca*, *htmr*, *inputsig*, *lcasmd*, *plcrthr, plctthr, tcamd*, *unequiposig*, *vcat*, *vmode*, *vpm*, *wtr*.

Moreover the parameter *istd* of all existing VCG ports are set to the value being provided in the ED-EQPT(cp) command.

The parameter value changes at the VCG ports are notified via ED-VCG notifications.

Change of Interface Standard - Interface standard change on GE10PL1 pack, update affected entities

A change of istd shall set the following parameters of all VCG tributaries on the pack to their default values:

- *brstintvl*, *brstthr*, *degthr*, *dexcthr*, *hptimais*, *hptimmoninmbs*, *outmbs*, *pmhpfeses*, *pmhpneses*, *ptca*, *ptrcrfmt*, *ptrcwfmt*, *tmonmd*, *trc*, *sdsfmode*, *tcamd*.

The parameter value changes at the VCG tributaries are notified via ED-VCGTRIB notifications, using the modifier for the new interface standard.

Change of TransMUX capacity

Update affected entities – TransMUX capacity unequal 0

The parameter *tmuxcap* can be set with the ED-EQPT(cp) command. Changing this parameter to a different value than 0 shall create following entities:

1. 8 instances of port TMAT3 is created. This are reported by means of ENT-TMAT3 notification for port.
2. 48 instances of port TMST3 is created. This are reported by means of ENT-TMST3 notification for port.
3. 1 STS1 path termination is created for each of the previously created TMST3 ports. This are reported by means of ENT-PTF-TMSSTS1 notifications.

4. 1 STS1 path termination is created for each of the previously created TMAT3 ports. This are reported by means of ENT-PTF-TMASTS1 notifications.
5. 28 DS1 NIMs are created for each of the previously created TMAT3 ports. This are reported by means of ENT-LONIM-DS1 notifications.
6. 28 VT1.5 PTFs are created for each of the previously created TMST3 ports. This are reported by means of ENT-LOPTF-TMSVT notifications.

Update affected entities – TransMUX capacity equal 0

The parameter *tmuxcap* can be set with the **`ED-EQPT(cp)`** command. Changing this parameter to 0 shall delete following entities:

1. 28 VT1.5 PTFs are deleted for each of the existing TMST3 ports. This are reported by means of **`DLT-LOPTF`** notifications.
2. 28 DS1 NIMs are deleted for each of the existing TMAT3 ports. This are reported by means of **`DLT-LONIM`** notifications.
3. 1 STS1 path termination is deleted for each of the existing TMAT3 ports. This are reported by means of **`DLT-PTF`** notifications.
4. 1 STS1 path termination is deleted for each of the existing TMST3 ports. This are reported by means of **`DLT-PTF`** notifications.
5. 48 instances of port TMAT3 is deleted This are reported by means of **`DLT-TMAT3`** notification for port.
6. 48 instances of port TMST3 is deleted This are reported by means of **`DLT-TMST3`** notification for port.

Group Notification for VT1.5 PTFs and DS1 NIMs

For efficiency reasons the notifications for VT1.5 PTFs and DS1 NIMs are sent out as a single notification (so-called group notification) using a range over the aid.

That means:

- the Simple_TMuxSLONumAID field in the TMuxSLONumAID is set to: 1&&-28
- the Simple_DS1NumAID field in the DS1NumAID is set to: 1&&-28

Example:

The AID field in a group notification for a tributaries related to synchronous and asynchronous adjunct ports of a TransMUX card in slot 1 looks like:

- DS1 NIM: 1-1-#-#-1-tma1-1-1&&-28
- VT1.5 PTF: 1-1-#-#-1-tms1-1-1&&-28

Implicit disabling of PM Monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. Stop PM counting
2. All current bins related to the PM monitor is deleted
3. All history bins related to the PM monitor is deleted

The term PM monitor has the same semantics as the term PM package. The number of PM monitors is determined by means of the number of ’pm enable attributes’.

PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts:

- The system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Change of ES64U capacity

Update affected entities - ES64U capacity.

No transmission entities shall be created or deleted by changing the ES64U capacity.

Example input/output

```
ED-EQPT:UNITE-NE:1-1-#-#-1-cp:123456:::eqptpn=NOALARM;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-EQPT`** command.

Validation rule: Validate ASAP profile

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: Check whether VLAN lists are empty

If *nvmname* is GE1 and the parameter *vlantagmd* is set to BCMD it is checked whether for the specified slot:

- the static VLAN lists (vvidls) of all EPORT entities are empty
- the static VLAN lists (vvidls) of all VCG entities are empty

- the default VLAN Id (dftvlan) of all EPORT entities is set to unavailable
- the default VLAN Id (dftvlan) of all VCG entities is set to unavailable.

Error code:

`SNVS – Status, Not in Valid State, static VLAN configuration not cleared`

Validation rule: Check whether LOXC circuit pack is member of an equipment protection group

If an attempt is made to change *loswcap* to 0, the circuit pack must not be member of an equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: Check whether LOXC circuit pack is in use

If an attempt is made to change *loswcap* to 0 while LO cross-connections or conversion cross-connections or LO substructure exist at one or more HO tributaries in the system, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Check number of active LOXC packs

Not more than 1 (2 in case of equipment protection groups) LOXC packs with *loswcap* = 15 may exist in the system.

Error code:

`SSRE – Status, System Resources Exceeded, allowed number of active LOXC packs exceeded`

Validation rule: LO switch capacity setting only for LOXC in worker slots

If an attempt is made, to change *loswcap* for an LOXC pack which is not in a dedicated equipment protection worker slot, then the command is denied.

Error code:

`IDNV – Input, Data Not Valid, invalid LO switch capacity for LOXC in this slot`

For the LOXC equipment protection dedicated pairs of worker and protection slots are defined. Only for the worker slots *loswcap* can be set away from 0. During creation of the protection group, *loswcap* of the protection LOXC is set autonomously.

Validation rule: LO switch capacity allowed for LOXC type

If an attempt is made to change *loswcap* to a value which is not supported by the LOXC type, then the command is denied.

Error code:

`IDNV – Input, Data Not Valid, invalid LO switch capacity for the LOXC type`

Validation rule: LO switch capacity allowed for both LOXC in an equipment protection group

If an attempt is made to change *loswcap* to a value which is not supported by both LOXC in the LOXC equipment protection group, then the command is denied.

Error code:

`IDNV – Input, Data Not Valid, invalid LO switch capacity for the LOXC type`

Validation rule: Only decreasing of LO switch capacity to 0 G allowed

If an attempt is made to decrease *loswcap* to another value than 0, then the command is denied.

Error code:

`IDNV – Input, Data Not Valid, invalid change of LO switch capacity`

Validation rule: No LO switch capacity setting without XC

If an attempt is made to change *loswcap* away from 0 for a LOXC pack and no XC is provisioned, then the command is denied.

Error code:

`SROF – Status, Requested Operation Failed,, invalid LOXC switch capacity without XC`

Validation rule: Check number of active TransMUX packs

Not more than 6 worker TransMUX packs with *tmuxcap* > 0 may exist in the system.

Error code:

`SSRE – Status, System Resources Exceeded, allowed number of active TransMUX packs exceeded`

Validation rule: Check whether TransMUX circuit pack is member of an equipment protection group

If an attempt is made to change *tmuxcap* to 0, the circuit pack must not be member of an equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: Check whether TransMUX circuit pack is in use

If an attempt is made to change *tmuxcap* to 0 while HO or LO cross connections exist at the TransMUX, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, HO or LO cross-connections still exist`

Validation rule: TransMUX capacity setting not for TransMUX in protection functionality

If an attempt is made, to change *tmuxcap* for a TransMUX pack which is the protecting pack in a equipment protection, then the command is denied.

The slots, where the change of *tmuxcap* is allowed, are defined in SRD_Configuration.

Error code:

`IDNV – Input, Data Not Valid, invalid TransMUX switch capacity for TransMUX in this slot`

Validation rule: No TransMUX capacity setting without XC

If an attempt is made, to change *tmuxcap* away from 0 for a TransMUX pack and no XC is provisioned, then the command is denied.

Error code:

`SROF – Status, Requested Operation Failed,, invalid TransMUX switch capacity without XC`

Validation rule: No ES64U capacity setting without XC

If an attempt is made, to change *es64ucap* away from 0 for a ES64U pack and no XC is provisioned, then the command shall be denied.

Error code:

`SROF – Status, Requested Operation Failed, invalid ES64U switch capacity without XC`

Validation rule: Check number of active ES64U packs

Not more than one ES64U packs with *es64ucap* > 0 may exist in the system.

Error code:

`SSRE – Status, System Resources Exceeded, allowed number of active ES64U packs exceeded`

Validation rule: Check whether ES64U circuit pack is member of an equipment protection group

If an attempt is made to change *es64ucap* to 0, the circuit pack must not be member of an equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: Check whether ES64U circuit pack is in use

If an attempt is made to change *es64ucap* to 0 while HO cross connections, test access sessions or loopbacks exist at the ES64U , then the command shall be denied.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check whether DCC Channel available for ES64U circuit pack

If an attempt is made to change *es64ucap* to 20 while no more DDC-M channel is available in the NE , then the command shall be denied.

Error code:

`SSRE – Status, System Resources Exceeded,  no free DDC-M channel available for ES64U`

Validation rule: ES64U capacity setting not for ES64U in protection functionality

If an attempt is made, to change *es64ucap* away from 20 for a ES64U pack which is the protecting pack in a equipment protection, then the command shall be denied.

Error code:

`IDNV – Input, Data Not Valid, invalid ES64U switch capacity for ES64U in this slot`

Validation rule: Auto-encapsulation must be enabled

If an attempt is made to change *es64ucap* to 20 while the auto-encapsulation functionality *autoencap* is not enabled, then the command shall be denied.

Error code:

`SNVS – Status, Not in Valid State,  Auto-Encapsulation must be switched on`

Validation rule: dcn_ipad must be configured

If an attempt is made to change *es64ucap* to 20 while the configured *dcn_ipad* address is not configured, then the command shall be denied.

Error code:

`SNVS – Status, Not in Valid State,  DCN IP Address is not configured`

Validation rule: dcn_ipad must be unique in the system

If an attempt is made to change *es64ucap* to 20 while the configured *dcn_ipad* address is already used in the system, then the command shall be denied.

Error code: `IDNV – Input, Data Not Valid, IP Address already used`

Validation rule: ESdcn_ipad setting not for card in protection functionality

If an attempt is made, to change *dcn_ipad* for a pack which is the protecting pack in a equipment protection, then the command shall be denied.

Error code: `SNVS – Status, Not in Valid State, Pack is protecting pack`

Validation rule: No interface standard change on GE10PL1 pack with existing cross-connection

Check that the interface standard is not changed on a GE10PL1 pack if there is a cross-connection existing at a VCG tributary of any VCG port on the pack.

Error code: `IDNV – Input, Data Not Valid, Cross-connections still existing`

Validation rule: No interface standard change on GE10PL1 pack with existing loopbacks

Check that the interface standard is not changed on a GE10PL1 pack if there is a cross-connection loopback existing at a VCG tributary of any VCG port on the pack.

Error code: `IDNV – Input, Data Not Valid, a loopback is existing`

Validation rule: No ONNS for interface standard change on GE10PL1 pack

Check that the interface standard is not changed on a GE10PL1 pack if ONNS functionality is available.

This means for any VCG on the pack that:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: No interface standard change on GE10PL1 pack with existing test access session

Check that the interface standard is not changed on a GE10PL1 pack if there is a test access session existing at a VCG tributary of any VCG port on the pack.

Error code:
`SNVS – Status, Not in Valid State, a test access session exists`

Related TL1 messages

- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
- "RTRV-EQPT (CP): Retrieve equipment (circuit pack)" (p. 3-2375)
- "DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)

References

GR-199-CORE

ED-EQPT (CP) notification

Output format

Syntax: **`ED-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of an equipment.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for circuit pack in any slot
   - LoxcAID::=UnivSlotAID-loxc for LOXC pack in universal slots

   The keyword **`usall`** is not allowed for Simple_UnivSlotAID.
   The keyword **`xcall`** is not allowed for Simple_XCSlotAID.
   The keyword **`ctlall`** is not allowed for Simple_CTLSlotAID.
   The keyword **`all`** is not allowed for Simple_SlotAID.
   PowerAID for power interface: The keyword **`piall`** is not allowed for PowerAID.
   EciAID for ECI interface The keyword **`eciall`** is not allowed for EciAID.
   TimingInterfaceAID for TI Interface board The keyword ″tiall″ is not allowed for TimingInterfaceAID.

Spec block parameters

1. **`dcn_ipad`** – DCN IP address for CLI/SNMP
   DCN IP address for CLI/SNMP management of the card.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255.
   The values 0.0.0.0, 1.1.1.1, 127.0.0.1, and 224.0.0.0 to 255.255.255.255 are excluded.
   Default: Empty string.
2. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the circuit pack. This parameter is reported for a CpAID, PowerAID, EciAID, TimingInterfaceAID and LoxcAID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.

3. **`eqptprotpn`** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT for the equipment.
   Option: This parameter is only reported for a CpAID, LoxcAID or PowerAID.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen ("-") is allowed in the name additionally.
   Default: DEFAULT
4. **`es64ucap`** – ES64U capacity
   Switch capacity of the ES64U pack.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0 – ES64U in suspended state, initial value after ES64U CP creation.
   - 20 – ES64U active or member of equipment protection group, capacity 20 Gbit/s.

   Default: 0.
5. **`vlantagmd`** – VLAN Tagging Mode
   VLAN tagging mode of the circuit pack. This parameter is relevant only for circuit packs of type GE1 (i.e. nvmname = GE1).
   Type: value set.
   Range:
   - BCMD (default), for the backward compatible mode
   - 802.1tag, for the VLAN tag based mode according to IEEE802.1Q.
6. **`loswcap`** – LOXC switch capacity
   Switch capacity of the LOXC pack. This parameter is relevant only for circuit packs of type LOXC, thus the parameter shall be omitted for other packs.
   Type: value set.
   Range, if *nvmname* = LOXC
   - 0: LOXC in suspended state, initial value after LOXC CP creation
   - 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s.

   Range, if *nvmname* = LOXC40G2S or LOXC40G3S:
   - 0: LOXC in suspended state, initial value after LOXC CP creation
   - 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s
   - 40: LOXC active or member of equipment protection group, switch capacity 40 Gbit/s

   Default: 0.

7. **`lostd`** – LO tributary standard
   LO tributary standard supported by the LOXC pack. This parameter is relevant only for circuit packs of type LOXC (i.e. *nvmname* = LOXC).
   Type: value set.
   Range:
   - SDH: LOXC supports SDH LO tributaries (VC3, VC12)
   - SONET: LOXC supports SONET LO tributaries (VT1.5)
   - UNDEFINED (default): no LO tributaries (loswcap = 0).

   LO VC3 tributaries provided with lostd=SDH support LO cross-connections and AU3/AU4 conversion cross-connections.
8. **`tmuxcap`** – TransMUX capacity
   Switch capacity of the TransMUX pack.
   This parameter is relevant only for circuit packs of type TransMUX (that means, *nvmname* = TransMUX), thus the parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0: TransMUX in suspended state, initial value after TransMUX CP creation.
   - 2G5: TransMUX active or member of equipment protection group, capacity 2.5 Gbit/s

   Default: 0
9. **`istd`** – Interface Standard of the Circuit Pack
   Interface standard supported by the circuit pack. This parameter is supported on GE10PL1 and ES64U circuit packs. On the circuit packs, the parameter *istd* defines the interface standard of the virtual concatenation groups.
   Option: If the parameter *istd* is not supported, then the parameter is omitted.
   Type: Value set.
   Range:
   - for GE10PL1: SDH or SONET
   - for ES64U: SONET or UNDEFINED (ES64U not used, *es64ucap* = 0)

   Default:
   - for GE10PL1: The default is derived from the system parameter *istddflt*.
   - for ES64U: UNDEFINED.

Example output

```
ED-EQPT:UNITE-NE:1-1-#-#-1-cp:123456:::eqptpn=NOALARM;

   UNITE-NE 00-11-14 08:00:00
M  123456 COMPLD
;

    UNITE-NE 00-11-14 08:00:30
A   001 REPT DBCHG
```

```
    "TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-1-cp:::eqptpn=NOALARM"
;
```

ED-EQPT (shelf): Edit equipment (shelf)

The **ED-EQPT** command modifies the provisioning values associated with the shelf.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no.

Input format

Command syntax: **ED-EQPT**:*tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of the shelf/subrack.
   Type: AID
   Range: ShelfAID. The keyword **all** is allowed on bay AID and shelf AID level.
   Default: 1-1.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **eqptpn** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
2. **eqptprotpn** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
3. **ssoride** – SS Bits Overrides
   This parameter determines the value to be inserted into the SS bits of the pointer of every AU-N or STS-N signal in the output direction of every SDH or SONET port in the shelf.

The terms *override*, *enable* and *disable* are used since:

- the default value of the SS bits in 00
- the parameter allows you to diverge from that default.

Type: Alphanumeric value set.
Range: ENABLE insert value 00 into SS bits, DISABLE insert value 10 into SS bits.
Default: DISABLE.

Output format

If the **ED-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The following attribute of the shelf entity is updated due to the command **ED-EQPT**:

- *eqptpn* as specified in the command input
- *eqptprotpn* as specified in the command input.

The modification is reported by means of an **ED-EQPT** notification for the shelf.

Example input/output

```
ED-EQPT:UNITE-NE:1-1:123456:::eqptpn=Nuernberg-1;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to RTRV-EQPT (shelf) command.

Validation rule: Check for ASAP profile existence

When no ASAP profile with the name specified by *eqptpn* or *eqptprotpn* does exist the following error message is returned.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Related TL1 messages

- "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400)
- "DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
- "RTRV-STATE-EQPT (system): Retrieve equipment state (system)" (p. 3-2845)

References

GR-199-CORE

ED-EQPT (shelf) notification

Output format

Syntax: **ED-EQPT**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   The AID of the shelf/subrack.
   Type: AID
   Range: ShelfAID. The keyword **all** is not allowed on bay AID and shelf AID level.
   Default: 1-1.

Spec block parameters

1. **eqptpn** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
2. **eqptprotpn** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
3. **ssoride** – SS Bits Overrides
   This parameter determines the value to be inserted into the SS bits of the pointer of every AU-N or STS-N signal in the output direction of every SDH or SONET port in the shelf.

   The terms *override*, *enable* and *disable* are used since:

   - the default value of the SS bits in 00
   - the parameter allows you to diverge from that default.

   Type: Alphanumeric value set.
   Range: ENABLE insert value 00 into SS bits, DISABLE insert value 10 into SS bits.
   Default: DISABLE.

Example output

```
ED-EQPT:UNITE-NE:1-1:123456:::eqptpn=Nuernberg-1;

   UNITE-NE 00-11-20 08:00:00
M  123456 COMPLD
;
```

```
   UNITE-NE  00-11-20  8:01:03
A  610  REPT  DBCHG
   ”Time=8-00-30,DATE=00-11-20:ED-EQPT:1-1::eqptpn=Nuernberg-1”
;
```

ED-EQPT (slot): Edit equipment (slot)

The **`ED-EQPT`** command modifies the provisioning values associated with slot.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **`ED-EQPT`***:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   The AID of a slot.
   Type: AID
   Range:
   - SlotAID
   - UnivSlotAID
   - XCSlotAID
   - CTLSlotAID.

   The keyword **`usall`** is allowed for Simple_UnivSlotAID.
   The keyword **`xcall`** is allowed for Simple_XCSlotAID.
   The keyword **`ctlall`** is allowed for Simple_CTLSlotAID.
   The keyword **`all`** is allowed for Simple_SlotAID.
   The keyword **`all`** is not allowed on bay/shelf level.
3. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT on slot level
   Type: Alphanumeric string
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
   Note: *eqptpn* is used on system, shelf and circuit pack level for a profile of type EQUIP-UNPROT.
2. **`nvmname`** – Circuit Pack Name
   This parameter specifies the name of the provisioned circuit pack of the slot. If no circuit pack is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.
3. **`nvmqual`** – Circuit Pack Qualifier
   This parameter specifies the qualifier of the provisioned circuit pack of the slot. If no circuit pack is provisioned the value UNDEF is reported.
   Type: value set.
   Range: 7 characters.
   The allowed values are:
   - UNDEF (default)
   - EXTD
   - all values as defined in SRD_ConfigurationManagement

   Note: The TL1-representation of the functional qualifier follows the following rules:
   - alphanumeric characters or the special character '-' only
   - special characters like '/' and '.' to be omitted.

Output format

If the **`ED-EQPT`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Generic behavior

The command modifies the provisioned circuit pack for a specified slot. The spec block parameters of the slot entity as specified in the command input is updated due to that operation.

The modification is reported by means of ED-EQPT notification for the slot.

OP10D reprovisioning

Impact on sockets when changing OP10D/PAR2 to an OP10D/PAR2XFP

If the command is used to change the provisioned circuit pack in the slot from an OP10D/PAR2 to an OP10D/PAR2XFP, then the optical module type for provisioned sockets on the pack is set to OMX10/10KM1. Unprovisioned sockets shall remain unprovisioned For each socket on the pack, an ED-EQPT notification without spec block parameters is sent.

Impact on sockets when changing OP10D/PAR2XFP to an OP10D/PAR2

If the command is used to change the provisioned circuit pack in the slot from an OP10D/PAR2XFP to an OP10D/PAR2, then the optical module type for provisioned sockets on the pack is set to OM10/1.5IR1. Unprovisioned sockets shall remain unprovisioned.

For each socket on the pack an ED-EQPT notification without spec block parameters is sent. The setting of the provisioned optical module type to a default module avoids the situation that an optical module is provisioned, which is not allowed for the new circuit pack type. This may lead to a module mismatch, which has to be corrected manually. The empty ED-EQPT notification for the sockets indicates the change of the socket type. With the change of the socket type all previous inventory parameter values for the sockets become invalid.

Example input/output

```
ED-EQPT:UNITE-NE:1-1-#-#-1:123456:::nvmname=OP10,nvmqual=1.5LR1;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ED-EQPT** command.

Validation rule: Check whether CP is acceptable

Check of the circuit pack type which is allowed for the slot.

Consistency is checked at two levels:

- between *actlname*/*actlqual* with *nvmname*/*nvmqual* at system level
- between *actlname* with *nvmname* at slot level.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized CP for this slot`

Validation rule: Check whether system is maintenance condition

Preprovisioning is not allowed, when the system in maintenance condition (see “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400)).

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Check for ASAP profile existence

When no ASAP profile with the name specified by *eqptpn* exists the following error is returned.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: Check exchangeability rules for optical packs

When the currently provisioned circuit pack is replaced by a new one by specifying the parameters *nvmname* and *nvmqual* the following exchangeability rules are checked whether all are fulfilled:

- the currently provisioned CP as well as the new one is an optical pack
- the holder state of the addressed slot is empty OR a pack with the specified *nvmname* and *nvmqual* is already plugged in
- the number of ports of both CPs is identical
- the transmission capacity of both CPs is identical.
- the current port configuration concerning forward error correction is supported by the new CP.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized CP for this slot`

Validation rule: Check exchangeability rules for electrical packs

When the currently provisioned circuit pack is replaced by a new one by specifying the parameters nvmname and nvmqual the following exchangeability rules are checked whether all are fulfilled:

- the currently provisioned CP as well as the new one is an electrical pack
- the holder state of the addressed slot is empty OR a pack with the specified nvmname and nvmqual is already plugged in.
- the number of ports of both CPs is identical
- the transmission capacity of both CPs is identical.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized CP for this slot`

Validation rule: Check exchangeability rules for XC packs

When the currently provisioned circuit pack is replaced by a new one by specifying the parameters nvmname and nvmqual the following exchangeability rules are checked whether all are fulfilled:

- the currently provisioned CP as well as the new one is a cross-connect pack AND
- the holder state of the addressed slot is empty OR a pack with the specified *nvmname* and *nvmqual* is already plugged in.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized CP for this slot`

Validation rule: Check exchangeability rules for LOXC packs

When the currently provisioned circuit pack is replaced by a new one by specifying the parameters nvmname and nvmqual the following exchangeability rules are checked whether all are fulfilled:

- the currently provisioned CP as well as the new one is an LOXC pack AND
- the holder state of the addressed slot is empty OR a pack with the specified *nvmname* and *nvmqual* is already plugged in.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized CP for this slot`

Related TL1 messages

- "RTRV-EQPT (slot): Retrieve equipment (slot)" (p. 3-2386)
- "DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
- "RTRV-STATE-EQPT (slot): Retrieve equipment state (slot)" (p. 3-2839)
- "RTRV-ACTLCP: Retrieve acceptable circuit pack list" (p. 3-2222).

References

GR-199-CORE

ED-EQPT (slot) notification

Output format

Syntax: **`ED-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Description: The AID of a slot.
   Type: AID
   Range: SlotAID
   The keyword **`usall`** is not allowed for Simple_UnivSlotAID.
   The keyword **`xcall`** is not allowed for Simple_XCSlotAID.
   The keyword **`ctlall`** is not allowed for Simple_CTLSlotAID.
   The keyword **`all`** is not allowed for Simple_SlotAID, Simple_UnivSlotAID, Simple_XCSlotAID, Simple_CTLSlotAID.
   The keyword **`all`** is not allowed on bay/shelf level.

Spec block parameters

1. **`actlname`** – Acceptable Circuit Pack Name
   This parameter is reported by itself The line is repeated for all groups of acceptable circuit packs in the slot. This list specifies the groups of circuit packs which can be provisioned for that slot. The detailed information about the groups can be requested via RTRV-ACTLCP.
   Type: value set.
   Range: 12 character.
2. **`nvmname`** – Circuit Pack Name
   This parameter specifies the name of the provisioned circuit pack of the slot. If no circuit pack is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.
3. **`nvmqual`** – Circuit Pack Qualifier
   This parameter specifies the qualifier of the provisioned circuit pack of the slot. If no circuit pack is provisioned the value UNDEF is reported.
   Type: value set.
   Range: 7 characters.

   The allowed values are:

   - UNDEF (default)
   - EXTD
   - all values as defined in SRD_ConfigurationManagement

Note: The TL1-representation of the functional qualifier follows the following rules:

- alphanumeric characters or the special character ’-’ only
- special characters like ’/’ and ’.’ to be omitted.

4. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT on slot level
   Type: Alphanumeric string
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
   Note: *eqptpn* is used on system, shelf and circuit pack level for a profile of type EQUIP-UNPROT.
5. **`name`** – Circuit Pack Name
   Circuit Pack Name as stored in the EEPROM of the replaceable unit.
   Type: value set.
   Range: up to 12 characters.
   Default: empty string ″″
   The default is reported on CP plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \″\″.
6. **`qual`** – Circuit Pack Qualifier
   Circuit Pack Qualifier as stored in the EEPROM of the replaceable unit.
   Type: value set.
   Range: up to 7 characters.
   Default: empty string ″″
   The default is reported on CP plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \″\″.
7. **`app`** – Apparatus Code
   Apparatus Code as stored in the EEPROM of the replaceable unit.
   Type: Alphanumeric string.
   Range: up to 9 characters.
   Default: empty string ″″
   The default is reported on CP plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \″\″.
8. **`ssn`** – Series Number
   Series Number as stored in the EEPROM of the replaceable unit.
   Type: value set.
   Range: up to 6 characters.
   Default: empty string ″″
   The default is reported on CP plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \″\″.
   Note: A dash (-) is used instead of a colon (:) for the series number in TL1.

9. **`clei`** – Common Language Equipment
   Common Language Equipment as stored in the EEPROM of the replaceable unit.
   Type: value set.
   Range: 10 characters.
   Default: empty string ″″
   The default is reported on CP plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \″\″.
10. **`eci`** – Equipment Catalog Item
    Equipment Catalog Item as stored in the EEPROM of the replaceable unit.
    Type: value set.
    Range: 6 digits.
    Default: empty string ″″
    The default is reported on CP plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
11. **`sln`** – Serial Number
    Serial number as stored in the EEPROM on the circuit pack to identify the replaceable unit and indicate the date and place of manufacture.
    Type: Alphanumeric string.
    Range: 12 characters.
    Default: empty string ″″
    The default is reported on CP plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.

Example output

The following examples shows the case after pre-provisioning of slot 1 for an OP2G5/13SR4 pack:

```
ED-EQPT:UNITE-NE:1-1-#-#-1:123456:::nvmname=OP2G5,nvmqual=13SR4;

   UNITE-NE 00-11-14 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-1::nvmname=OP2G5,nvmqual=13SR4″
;
```

The following examples shows a case after auto-provisioning of slot 1 with an OP2G5/13SR4 pack (note: it is possible that change of *nvmname*, *nvmqual* and change of inventory data is reported in two notifications):

```
   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-1::nvmname=OP2G5,nvmqual=13SR4″
```

```
   UNITE-NE 00-11-14 08:00:31
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-
      1::name=OP2G5,qual=13SR4,app=KFA12,ssn=1-
      1,clei=WMTBJDGAAA,eci=276912,sln=123456789012″
;
```

The following examples shows a case after plug-out of the OP2G5/13SR4 pack from slot 1:

```
   UNITE-NE 00-11-14 08:00:31
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-
      1::name=\″\″,qual=\″\″,app=\″\″,ssn=\″\″,clei=\″\″,eci=\″\″,sln=\″\″″
;
```

The following examples shows the case after auto-provisioning of an OP40 into slot 4 and the effect on slot 1:

```
   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-1::nvmname=OP40,nvmqual=EXTD″
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-1::actlname=\″\″″
;
```

ED-EQPT (socket): Edit equipment

The **ED-EQPT** command shall modify the provisioning values associated with socket.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no.

Input format

Command syntax: **ED-EQPT***:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **eqptpn** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT on socket level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen ("-") is allowed in the name additionally.
   Default: DEFAULT.
2. **nvmname** – Provisioned Module Name
   This parameter specifies the name of the provisioned module of the socket. If no module is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.

3. **nvmqual** – Provisioned Module Qualifier
This parameter specifies the qualifier of the provisioned module of the slot. If no module is provisioned the default value is reported..
Type: value set.
Range: 7 characters.
Default: UNDEF.
Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:
- alphanumeric characters or the special character '-' only
- special characters like '/' and '.' to be omitted.

Output format

If the **ED-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the provisioned module for a specified socket. The spec block operation. The modification is reported by means of ED-EQPT notification for the slot.

Example input/output

This example shows an ED-EQPT command which reprovisions the socket and assigns a new ASAP:

```
ED-EQPT:UNITE-NE:1-1-#-#-5-sc2:123456:::nvmname=OM2G5,nvmqual=15LR1,eqptpn=PROFILE1;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-EQPT** command.

Validation rule: Check whether system is in maintenance condition

Preprovisioning is not allowed, when the systemis in maintenance condition. (See "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).)

Error code:
`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Check for ASAP profile existence

When no ASAP profile with the name specified by *eqptpn* exists the following error is returned.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: Socket must already be provisioned

If the *nvmname* of the pack the socket belongs to is **not equal to** GE10PL1 the following rule is applicable:

When re-provisioning a module, i.e. changing *nvmname*/*nvmqual*, it is checked whether the socket is already provisioned (i.e. *nvmname*/*nvmqual* is different from UNDEFINED/UNDEF).

Error code: `SNVS – Status, Not in Valid State, socket is not provisioned`

Validation rule: Exchangeability I: Equipage status and inventory data of module must be accessible

If the socket is already provisioned (i.e. *nvmname*/*nvmqual* is different from UNDEFINED/UNDEF) but the equipage status or inventory data of the module is not accessible (e.g. due to the parent pack being in recovery) re-provisioning of the socket is not allowed.

Error code: `SNVS – Status, Not in Valid State, unknown module equipage`

Validation rule: Exchangeability II: Invalid module plugged in

If

- the socket is already provisioned (i.e. *nvmname*/*nvmqual* is different from UNDEFINED/UNDEF)
- the socket is neither empty nor equipped with the new type of module being specified by *nvmname*/*nvmqual*

then re-provisioning of the socket is not allowed.

Error code:
`SNVS – Status, Not in Valid State, the module is not plugged out`

Validation rule: Exchangeability III: Invalid module plugged in

If

- the socket is already provisioned (i.e. *nvmname*/*nvmqual* is different from UNDEFINED/UNDEF),
- the socket is not empty
- the socket is equipped with the new type of module being specified by *nvmname*/*nvmqual*
- the new type of module being specified by *nvmname*/*nvmqual* is not in the list of acceptable modules (given by parameters *actlname*, *actlqual* of RTRV-EQPT(socket))

then re-provisioning of the socket is not allowed.

Error code:

`ENRE – Equipage, Not Recognized Equipage, unrecognized module for this socket`

Related TL1 messages

- "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
- "ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
- "RTRV-EQPT (socket): Retrieve equipment (socket)" (p. 3-2391)
- "RTRV-STATE-EQPT (socket): Retrieve equipment state " (p. 3-2842)

References

GR-199-CORE

ED-EQPT (socket) notification

Output format

Syntax: **`ED-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **`all`** is not allowed on bay/shelf level.

Spec block parameters

1. **`actlname`** – Acceptable Module Name
   This parameter is reported by itself together with actlmodqual. The line is repeated for all acceptable modules in the socket. This list specifies the modules which can be provisioned for that socket.
   Type: value set.
   Range: 12 characters.
2. **`actlqual`** – Acceptable Module Qualifier
   This parameter is reported by itself together with actlmodname. The line is repeated for all acceptable modules in the socket. This list specifies the modules which can be provisioned for that socket.
   Type: value set.
   Range: 7 characters.

   Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

   - alphanumeric characters or the special character ’-’ only
   - special characters like ’/’ and ’.’ to be omitted
3. **`app`** – Apparatus Code
   This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED. The 9 character *app* string is represented left aligned and padded on the right with ASCII spaces (20HEX) to a 12 character alphanumeric string.
   Type: Alphanumeric string
   Range: 12 characters alphanumeric string or empty string.
   Default: empty string ″″. The default is reported on module plug-out when the EEPROM is not readable or when the Apparatus Code value cannot be derived from the EEPROM content.. Note: In TL1 output direction the quotes have to escaped \″\″.

4. **`ccode`** – Commercial Code
   Commercial code + suffix as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
   Type: Alphanumeric string
   Range: up to 12 digits
   Default: empty string "". The default is reported on module plug-out or when the EEPROM is not readable. Note: In TL1 output direction the quotes have to escaped \"\".
5. **`clei`** – Common Language Equipment
   Common Language Equipment as stored in the EEPROM of the module.
   Type: value set.
   Range: 10 characters.
   Default: empty string "".
   The default is reported on module plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \"\".
6. **`compatibilitybyte`** – Compatibility Byte
   This parameter indicates the compatibility of the module with the system. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
   Type: Numeric string or text string
   Range: 0 … 255.
   Default: empty string "".
   The default is reported on module plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \"\".
7. **`connectortype`** – Connector Type
   This parameter indicates the type of connector at the module plugged into the socket. (E.g. 07 for LC). This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
   Type: Hexadecimal number or text string
   Range: 1 byte which is represented as 2 hexadecimal digits.
   Default: empty string "".
   The default is reported on module plug-out or when the EEPROM is not readable.
   Note: In TL1 output direction the quotes have to escaped \"\".
8. **`datecde`** – Date Code
   Vendor Date Code as stored in the EEPROM on the module. This parameter is kept for compatibility reasons only and is reported with the fixed value empty string "" for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.

Type: Text string.
Range: empty string ″″.
Default: empty string ″″.
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

9. **eci** – Equipment Catalog Item
Equipment Catalog Item as stored in the EEPROM of the module. This parameter is reported for a SocketAID, if the data in the EEPROM is stored according to the Small EEPROM Format.
Type: Value set.
Range: 6 digits.
Default: empty string ″″.
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

10. **eqptpn** – Equipment ASAP Profile Name
The name of the current ASAP profile of type EQUIP-PROT on socket level.
Type: Alphanumeric string.
Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
Default: DEFAULT.

11. **interchangemarker** – Interchangeability Marker
This parameter is derived from the EEPROM content on the module and reported for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: 6 characters.
Default: ″S1-1″.
The default is reported on module plug-out, when the EEPROM is not readable or when the Interchangeability Marker value is not available in the EEPROM.
Note: A dash (-) is used instead of a colon (:) for the interchangeability marker in TL1.

12. **name** – Module Name
Module Name as stored in the EEPROM of the module.
Type: value set.
Range: up to 12 characters.
Default: empty string ″″.
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

13. **nvmname** – Provisioned Module Name
This parameter specifies the name of the provisioned module of the socket. If no module is provisioned the default value is reported.
Type: value set.
Range: 12 character.
Default: UNDEFINED.

14. **`nvmqual`** – Provisioned Module Qualifier
    This parameter specifies the qualifier of the provisioned module of the slot. If no module is provisioned the default value is reported..
    Type: value set.
    Range: 7 characters.
    Default: UNDEF.

    Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

    - alphanumeric characters or the special character '-' only
    - special characters like '/' and '.' to be omitted.

15. **`physicalidentifier`** – Physical Identifier
    The Physical Identifier indicates the type of module plugged into the socket. (E.g. 03 for SFP). This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
    Type: Hexadecimal number or text string.
    Range: 1 byte which is represented as 2 hexadecimal digits.
    Default: empty string "".
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \"\".

16. **`prodid`** – Product Line Id
    Binary Compatibility + UNITE Serial Number Identifier as stored in the EEPROM on the module. This parameter is kept for compatibility reasons only and is reported with the fixed value empty string "", if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Text string
    Range: empty string ""
    Default: empty string "".
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \"\".

17. **`qual`** – Module Qualifier
    Module Qualifier as stored in the EEPROM of the module.
    Type: value set.
    Range: up to 7 characters.
    Default: empty string "".
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \"\".

18. **sln** – Serial Number
    Serial number as stored in the EEPROM on the module to identify the module and indicate the date and place of manufacture.
    Type: Alphanumeric string.
    Range: 12 characters.
    Default: empty string ″″.
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
19. **ssn** – Series Number
    Series Number as stored in the EEPROM of the module.
    Type: value set.
    Range: up to 6 characters.
    Default: empty string ″″.
    The default is reported on modlue plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
    Note: A dash (-) is used instead of a colon (:) for the series number in TL1.
20. **transceivercode** – Transceiver Code
    This parameter indicates the type of transceiver at the module plugged into the socket. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
    Type: Hexadecimal number or text string
    Range: 8 bytes where each byte is represented as 2 hexadecimal digits.
    Default: empty string ″″
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
21. **uniquenumber** – Lucent Unique Number
    Lucent Unique Number as stored in the EEPROM. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED. The Lucent Unique Number is represented right aligned and padded on the left with ASCII spaces (20HEX) to a 12 character numeric string.
    Type: Numeric String or text string
    Range: 12 characters alphanumeric string or empty string.
    Default: empty string ″″
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
22. **vendorname** – Vendor Name
    This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.

Type: Text string
Range: 16 characters
Default: empty string ″″
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

23. **`vendoroui`** – Vendor Company Identifier
This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Hexadecimal number or text string
Range: 3 bytes where each byte is represented as 2 hexadecimal digits.
Default: empty string ″″
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

24. **`vendorpartno`** – Vendor Part Number
This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for a the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: 16 characters
Default: empty string ″″
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

25. **`vendorrevlvl`** – Vendor Revision Level
Vendor Revision Level as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: 4 characters
Default: empty string ″″
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

26. **`vendorsln`** – Vendor Serial Number
Vendor Serial Number + WES Vendor Identifier as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: 20 characters
Default: empty string ″″
The default is reported on module plug-out or when the EEPROM is not readable.
Note: In TL1 output direction the quotes have to escaped \″\″.

27. **`wesserialno`** – WES Serial Number
    This parameter is read from the EEPROM on the module and reported for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Text string
    Range: 18 characters
    Default: empty string ″″
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.
28. **`wesvendorid`** – WES Vendor ID
    WES Vendor Identifier as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Text string
    Range: 4 characters
    Default: empty string ″″
    The default is reported on module plug-out or when the EEPROM is not readable.
    Note: In TL1 output direction the quotes have to escaped \″\″.

Example output

This example shows the ED-EQPT notification after reprovisioning of a socket:

```
   UNITE-NE 00-11-14 08:00:30
A 001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-5-
      sc2::nvmname=OM2G5,nvmqual=15LR1,eqptpn=PROFILE1″
;
```

This example shows the ED-EQPT notification reporting the inventory data after plugging in a module with data stored in Small EEPROM forma:

```
   UNITE-NE 00-11-14 08:00:30
A 001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-5-sc2::name=OM2G5,qual=15LR1,
ccode=109192278000,clei=WM3ILPEDAA,eci=255694,sln=00NB01002333,ssn=1-1″
;
```

This example shows the ED-EQPT notification reporting the inventory data after plugging in a modul with data stored in SFP EEPROM format:

```
   UNITE-NE 06-06-30 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=06-06-30:ED-EQPT:1-1-#-#-5-
      sc2::name=OM2G5,qual=CL61L1,app=\″\″,ccode=109496869,clei=WM1T37KBAA,compatibilitybyte=
;
```

This example shows the ED-EQPT notification reporting the not readable inventory data after plug out of a module with data stored in Small EEPROM format:

```
   UNITE-NE 00-11-14 08:00:30
A 001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-EQPT:1-1-#-#-5-sc2::name=\″\″,qual=\″\″,
comcde=\″\″,clei=\″\″,eci=\″\″,sln=\″\″,ssn=\″\″″
;
```

ED-EQPT (system): Edit equipment (system)

The **ED-EQPT** command modifies the provisioning values associated with system (NE) level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-EQPT***:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
The aid of the system itself.
Type: AID
Range: system. The usage of the keyword **all** on system level is not allowed.
Default: system
3. **ctag** – Correlation Tag
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **eqptpn** – Equipment ASAP Profile Name
The name of the current ASAP profile on the system level
Type: Alphanumeric string
Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
Default: DEFAULT
2. **istddflt** – Interface Standard Default
Defining the predominantly used interface standard for a port.
Type: Alphanumeric value set.
Range:
   - SONET
   - SDH

   Default: SDH

3. **`mondelay`** – Port Auto to Mon Delay
   The switch of the port mode from the value AUTO to the value MON is delayed by this timer. It requires that a valid signal is present for the set timer period prior to the port transitioning to monitored. If during the timer period the signal is interrupted, the time should reset and begin another period when a valid signal is once again presented.
   Type: Numeric string.
   Range:
   - 0 = no timer will be started
   - 1…480 = in steps of 1 representing a switch delay of the port mode in minutes.

   Default: 0
4. **`portclassdflt`** – Port Class Default
   Defines the default port class, which is used when a new port is created and xcsetup=SWISS. When xcsetup=TRADITIONAL this parameter is not evaluated.
   Type: Alphanumeric value set.
   Range:
   - TRADITIONAL
   - EDGE

   Default: TRADITIONAL.
5. **`synccrtc`** – System Synchronization Characteristic
   This parameter specifies whether the NE uses the SDH or SONET timing function for synchronization.
   Type: Alphanumeric value set.
   Range:
   - SONET
   - SDH.

   Default: SDH.
6. **`trbmddflt`** – Tributary Mode Default
   Defining the predominantly used tributary mode for a port.
   Type: Alphanumeric value set.
   Range:
   - FIXED
   - ADAPTIVE

   Default: FIXED.
7. **`swmaxcap`** – Maximum Switch Capacity
   This parameter indicates the maximum switch capacity of the system that the user intends to use.
   Type: value set.
   Range:
   - LXC160 (default), 160 GBit/s switch capacity
   - LXC320, 320 GBit/s switch capacity.

8. **`upsrportalarming`** – Enh. SA/NSA Detection for UPSR and SNCP
   The parameter specifies whether the Enhanced SA/NSA Detection for UPSR Rings and SNCP is switched on or off.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
9. **`xcsetup`** – Cross Connect Setup
   This parameter indicates whether the system supports the set-up of cross-connections via the ONNS application.
   Type: value set.
   Range: SWISS ONNS application is enabled; TRADITIONAL ONNS application is disabled.
   Default: TRADITIONAL.

Output format

If the **`ED-EQPT`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the current values for the system. The values of the attributes as listed above in section spec block parameters are updated as specified in the command input.

Change of the parameters *istddflt, trbmddflt* has no effect on the corresponding parameters for existing entities on port and tributary level, but only on entities which are created afterwards.

The modification is reported by means of an **`ED-EQPT`** notification for the system.

Dependencies *istddflt* and *trbmddflt*

If *istddflt* is changed to SDH the *trbmddflt* parameter is set to FIXED by the system.

Updated affected entities

See discussion in Chapter 10, "TL1 messages - synchronization" after change of *synccrtc* and leave of MCOND.

Behavior when setting of *xcsetup* from TRADITIONAL to SWISS or vice versa

The system must assure that the validation rules as listed in the Error responses section are met.

To set the system back from *xcsetup* = SWISS to *xcsetup* = TRADITIONAL it has to be ensured that no resources are in use by SWISS (ONNS).

The following steps are recommended to achieve this:

- identify all SDH/SONET ports with *portClass* = INNI or *portClass* = EDGE, then set *portControl* = LOCKOUT
- identify all GbE ports with *eportClass* = INNI or *eportClass* = EDGE, then set *eportControl* = LOCKOUT
- ensure that all NN paths which start, end, or traverse this NE are deleted or re-routed by the SWISS application via NN TL1 ENT/ED/DLT-PATH
- In exceptional cases, when the previous step is not possible: Identify all remaining cross-connections with *xconcls* = NN, then use ED-NNCRS to set *xconcls* = TRADITIONAL
- identify all SDH/SONET ports with *portClass* = INNI or *portClass* = EDGE, then set *portClass* = TRADITIONAL
- identify all GbE ports with *eportClass* = INNI or *eportClass* = EDGE, then set *eportClass* = TRADITIONAL
- identify all DCC channels with *scn_dccstat* = ENABLE, then set *scn_dccstat* = DISABLE
- identify all LANs with *osinode* = scnnode, then either set osinode to osinode{1..8} or DISABLE lanstat, ip_lanstat.

Dependency between setting of *xcsetup* and acceptable CP type list

By changing the value of *xcsetup* also the acceptable circuit pack type list changes.

Dependency between *xcsetup* and ONNS TL1 commands

While *xcsetup* = TRADITIONAL any ONNS TL1 command is rejected.

Update affected entities when changing *swmaxcap* from LXC160 to LXC320

When changing *swmaxcap* from LXC160 to LXC320 the acceptable circuit pack name list *actlname*:

- of all universal slots in the lower row shall change from an empty list to a non-empty list.
- of the XC slots 9 and 10 is modified such that it does not contain XC160 anymore.

The modification is reported by means of ED-EQPT notifications for the affected slot entities.

Update affected entities when changing *swmaxcap* from LXC320 to LXC640

When changing *swmaxcap* from LXC320 to LXC640 the acceptable circuit pack name list *actlname*:

- of universal slot n-1 shall change from an emtpy list to a non-empty list if currently a double density circuit pack is provisioned in slot n.
- of the XC slots 9 and 10 is modified such that it does not contain XC320 anymore.

The modification is reported by means of ED-EQPT notifications for the affected slot entities.

Update affected entities when changing *swmaxcap* from LXC160 to LXC640

When changing *swmaxcap* from LXC160 to LXC640 the acceptable circuit pack name list *actlname*:

- of all universal slots in the lower row shall change from an emtpy list to a non-empty list.
- of universal slot n-1 shall change from an emtpy list to a non-empty list if currently a double density circuit pack is provisioned in slot n
- of the XC slots 9 and 10 is modified such that it does not contain XC160 and XC320 anymore.

The modification is reported by means of ED-EQPT notifications for the affected slot entities.

Dependency *portclassdflt* with *xcsetup*

Change of the parameter *portclassdflt* shall not have effects on existing ports. The parameter *portclassdflt* shall automatically be set to TRADITIONAL, when *xcsetup* is set to TRADITIONAL.

Example input/output

```
ED-EQPT:UNITE-NE:system:123456:::istddflt=SONET;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;

ED-EQPT:UNITE-NE:system:123456:::portclassdflt=EDGE,xcsetup=SWISS;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-EQPT** command.

Validation rule: check for ASAP profile existence

When no ASAP profile with the name specified by *eqptpn* does not exist the following error is returned.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: check of the maintenance condition

System must be in MCOND for change of *synccrtc*.

Error code:
`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: compatibility *istddflt* and *trbmddflt*

Change of *trbmddflt* to ADAPTIVE while *istddflt*= SDH is not allowed.

Error code:
`SNVS – Status, Not in Valid State, incompatibility istddflt and trbmddflt`

Validation rule: Compatibility *xcsetup* and HW configuration

The system shall verify that ONNS capable HW is (pre-) provisioned during an attempt to set *xcsetup* = SWISS. Otherwise the following error is reported:

Error code: `SNVS – Status, Not in Valid State, no SwiSS capable HW`

Validation rule: Compatibility *xcsetup* and NVM size

The system shall verify that the installed NVM has a capacity > 256 Mbytes during an attempt to set *xcsetup* = SWISS. Otherwise the following error is reported:

Error code: `SNVS – Status, Not in Valid State, NVM size not sufficient`

Validation rule: Compatibility *xcsetup* and IP configuration for SCN

The system shall verify that the mandatory ASTN IP address is provisioned during an attempt to set *xcsetup* = SWISS. Otherwise the following error is reported:

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

See corresponding validation rules in “ENT-ULS: Enter upper layer stack” (p. 3-1944) for details.

Validation rule: Compatibility *xcsetup* and software state

The system verifies that the software is not in trial phase during an attempt to set xcsetup=SWISS. Otherwise the following error is reported:

Error code:
`SNVS – Status, Not in Valid State, SW generic is in trial phase`

Validation rule: Setting *xcsetup* back to TRADITIONAL

On an attempt to set *xcsetup* to TRADITIONAL it has to be ensured that all of the following rules are fulfilled:

- for all (pre-)provisoned SDH/SONET ports (i.e. ports which are identified via SPortAID) in the system holds *portClass* = TRADITIONAL, see RTRV/ED-rr (port)
- for all (pre-)provisoned GbE ports (i.e. ports which are identified via EPortAID) in the system holds *eportClass* = TRADITIONAL, see RTRV/ED-EPORT
- for all cross-connections in the system holds *xconcls* = TRADITIONAL, see RTRV/ED-CRS, ED-NNCRS
- no DCC channel shall have *scn_dccstat* = ENABLED, see RTRV/ENT-FECOM
- no LAN is assigned to the SCN with *osinode* = scnnode, see RTRV/ENT-FECOM-LAN.

Otherwise the following error is reported:

Error code: `SNVS – Status, Not in Valid State, resources in use by SwiSS`.

Validation rule: Do not accept ONNS TL1 commands when *xcsetup* = TRADITIONAL

The system shall DENY ONNS TL1 commands when *xcsetup* = TRADITIONAL.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: Do not accept setting *portclassdflt* to EDGE when *xcsetup* = TRADITIONAL

The system shall DENY setting the parameter *portclassdflt* to the value EDGE when xcsetup=TRADITIONAL.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: Maximum switch capacity according XC CP

If the *swmaxcap* parameter is set it is ensured that the provided switch capacity is not higher than the switch capacity of the provisioned XC circuit pack with the smallest switch capacity in the system.

Error code:

`SNVS – Status, Not in Valid State, switch capacity below provisioned XC CP switch capacity`

Validation rule: No decrease of maximum switch capacity

If the *swmaxcap* parameter is set it is ensured that the *swmaxcap* value in the **`ED-EQPT`** command is not lower than the actual provisioned swmaxcap value in the system.

Error code:

`IDNV – Input, Data Not Valid, switch capacity cannot be decreased`

Related TL1 messages

- “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400)
- “DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
- “RTRV-STATE-EQPT (system): Retrieve equipment state (system)” (p. 3-2845)
- “RTRV-NNCNFG: Retrieve ONNS configuration data” (p. 3-2555)
- “ED-NNCNFG: Modify ONNS system configuration” (p. 3-450)

References

GR-199-CORE

ED-EQPT (system) notification

Output format

Syntax: **`ED-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system. The usage of the keyword **`all`** on system level is not allowed.
   Default: system

Spec block parameters

1. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile on the system level
   Type: Alphanumeric string
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT
2. **`istddflt`** – Interface Standard Default
   Defining the predominantly used interface standard for a port.
   Type: Alphanumeric value set.
   Range:
   - SONET
   - SDH
   Default: SDH
3. **`mondelay`** – Port Auto to Mon Delay
   The switch of the port mode from the value AUTO to the value MON is delayed by this timer. It requires that a valid signal is present for the set timer period prior to the port transitioning to monitored. If during the timer period the signal is interrupted, the time should reset and begin another period when a valid signal is once again presented.
   Type: Numeric string.
   Range:
   - 0 = no timer will be started
   - 1…480 = in steps of 1 representing a switch delay of the port mode in minutes.
   Default: 0
4. **`portclassdflt`** – Port Class Default
   Defines the default port class, which is used when a new port is created and xcsetup=SWISS. When xcsetup=TRADITIONAL this parameter is not evaluated.
   Type: Alphanumeric value set.

Range:

- TRADITIONAL
- EDGE

Default: TRADITIONAL.

5. **synccrtc** – System Synchronization Characteristic
This parameter specifies whether the NE uses the SDH or SONET timing function for synchronization.
Type: Alphanumeric value set.
Range:
    - SONET
    - SDH.

    Default: SDH.
6. **trbmddflt** – Tributary Mode Default
Defining the predominantly used tributary mode for a port.
Type: Alphanumeric value set.
Range:
    - FIXED
    - ADAPTIVE

    Default: FIXED
7. **swmaxcap** – Maximum Switch Capacity
This parameter indicates the maximum switch capacity of the system that the user intends to use.
Type: Value set.
Range:
    - LXC160 160 GBit/s switch capacity
    - LXC320 320 GBit/s switch capacity
    - LXC640 640 GBit/s switch capacity

    Default: LXC160.
8. **upsrportalarming** – Enh. SA/NSA Detection for UPSR and SNCP
The parameter specifies whether the Enhanced SA/NSA Detection for UPSR Rings and SNCP is switched on or off.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
9. **xcsetup** – Cross Connect Setup
This parameter indicates whether the system supports the set-up of cross-connections via the ONNS application.
Type: value set.
Range: SWISS ONNS application is enabled; TRADITIONAL ONNS application is disabled.
Default: TRADITIONAL.

Example output

```
ED-EQPT:UNITE-NE:system:123456:::istddflt=SONET;

   UNITE-NE 00-11-14 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-14 08:00:50
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-14:ED-EQPT:system::istddflt=SONET”
;
```

ED-FECOM notification

Output format

The **ED-FECOM** notification has the following specific syntax:

ED-FECOM::*port*,*dcctype*:*spec_block*

Output parameters

Command parameters

1. **port**
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID.
   Range: SPortAID; the keyword **all** is not allowed.
2. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC of the *port* is affected by the command.
   Type: Alphanumeric value set.
   Range:
   - LINE
   - SECTION.

   Default:
   - SECTION if dccstat is set to ENABLE
   - no default (i.e. parameter is mandatory) if *scn_dccstat* or uni_dccstat is ENABLE.

Spec block parameters

1. **dccprottp**
   This parameter reports the protection type of the DCC channel. 0X1 means the port is not involved in a protection group or DCC protection switching is disabled. 1+1 means DCC protection switching is enabled: In these case only the active worker line is shown.
   Type: Alphanumeric value set.
   Range: 0X1 (default), 1+1.
2. **dccstat**
   Note: The default value of DISABLED is used when a DCC channel is created, this will be reported via the ENT-FECOM notification, or when a Transparent DCC Cross Connection is deleted, this will be reported via the ED-FECOM notification. The value TRANSPARENT is reported when a Transparent DCC Cross Connection is created. In this case all other spec_block parameters does not have any effect on operation of the system if reported. The other spec_block parameters does have only operational effect if the value is ENABLED.

This parameter indicates whether MCN communication over the DCC channel is enabled or if it is used in a Transparent DCC Cross Connection.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE (default), TRANSPARENT.
Note: Output responses with *dccstat* = DISABLE will only sent when either *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE or *enni_dccstat* = ENABLE, see also Detailed behavior description below.

3. **`lapdmode`**
This parameter indicates the type of LAPD service. Modes are AITS and UITS. Starting with R2.0 this parameter is also used to determine the service for the LinkID protocol.
If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
Type: Alphanumeric value set..
Range:
   - UITS (default)
   - AITS.
4. **`lapdrl`**
This parameter indicates the LAPD role. This parameter is also used to determine the service for the LinkID protocol.
If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
Type: Alphanumeric value set.
Range:
   - USER-SIDE (default)
   - NETWORK-SIDE.
5. **`osinode`**
Assignment of a DCC channel to an MCN node in the NE. The parameter is omitted when *dccstat* = ENABLE.
An NE can have up to 8 instances of the OSI stack. Node-1 is default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00). Nodes 2-8 can be created and configured, see command **`ENT-ULSDCC-L3`**.
Type: AID.
Range: osinode{1-8}

6. **`scn_dccstat`**
   This parameter indicates whether the DCC channel is used for communication over the SCN or is involved in a transparent DCC cross-connection.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT.
   Note: Output responses with *scn_dccstat* = DISABLE will only sent when either *dccstat* = ENABLE or *uni_dccstat* = ENABLE or *enni_dccstat* = ENABLE, see also Detailed Behavior Description below.
7. **`uni_dccstat`**
   This parameter indicates whether the DCC channel is used for UNI communication.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT.
   Note: Output responses with *uni_dccstat* = DISABLE will only sent when either *dccstat* = ENABLE or *scn_dccstat* = ENABLE or *enni_dccstat* = ENABLE, see also Detailed Behavior Description below.
8. **`enni_dccstat`**
   This parameter indicates whether the DCC channel is used for E-NNI communication.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT.
   Note: Output responses with *enni_dccstat* = DISABLE will only be sent when either *dccstat* = ENABLE or *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE, see also Detailed Behavior Description below.

Example output

```
   NE-UNITE 00-11-03 01:00:00
A  000 REPT DBCHG
   ”TIME=01-00-00,DATE=00-12-13:ED-FECOM::1-1-#-#-5-3,line:dccprottp=0X1”
;
```

ED-FETID: Edit far end target identifier

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No

Input format

Command syntax: **`ED-FETID`**:*tid*:[*aid*]:*ctag*::[*rid*],*spec_block;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **`ctag`** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.

Spec block parameters

1. **oldtid** – Old Target Identifier
   This parameter is the previous TID of a remote node on the ring which may be associated with one or more local cross-connections.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: 1 to 20 characters.
2. **newtid** – New Target Identifier
   The newtid parameter is the previous TID of a remote node on the ring which may be associated with one or more local cross-connections.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: 1 to 20 characters.

Output parameters

1. **sid** – Source Identifier
   This is the currently active name of the network element to which the command is addressed. The tid is a case insensitive character string.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: 1 to 20 characters.
2. **date** – Date
   Date when output message is generated. This has the format YY-MM-DD (year-month-day)
   Type: value set.
   Range: for YY 00 to 99, for MM 01 to 12, for DD 01 to 31
   TL1 messages use YY-MM-DD for the format of a date, meaning that the year only has two digits. This is consistent with Telcordia requirements. Internally, however, the system does use a 4-digit year. When the year appears in a response, the internal 4-digit year is truncated to two digits. When the year appears as an input parameter, for example in the **ED-DAT** command, years in the range 90-99 will represent 1990-1999, and years in the range 00-37 will represent 2000-2037.
   Default:00-01-01
3. **time** – Time
   Time when output message is generated. This has the format HH:MM:SS (hours:minutes:seconds).
   Type: Value set.
   Range: for HH 00 to 23, for MM 00 to 59, for SS 00 to 59.
   Default: 00:00:00
4. **M** – Message
   M indicates that the output message is generated in response to a manual command.
   Type: Value set.
   Range: M

5. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: Non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
6. **`IP`** – In Progress
   IP is sent in response to all commands with the exception of commands that result in error code output.
   Type: Value set.
   Range: IP
7. **`RL`** – Repeat Later
   The requested command cannot be executed due to unavailable system resources. The command may be entered again later.
   Type: Value set.
   Range: RL
8. **`COMPLD`** – Completion Code
   It indicates that the command has been completed.
   COMPLD for successful command execution: For a command that is decomposed into a set of atomic commands a COMPLD completion code means that the command was successful for all atomic commands (e.g. AID ranges, group alias).
   COMPLD for no changes: If a parameter has a current value, and a provisioning command attempts to reprovision the parameter with a value that is the same as its current value, the command will return a COMPLD completion code.
   Type: Value set.
   Range: COMPLD.

Output parameters

See definition in output parameters section in “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **`ED-FETID`** command is used to update the TID of a remote node on the ring which may be associated with one or more local cross-connections. The command can only be applied when automatic ring discovery is disabled. The network provider would then have to issue this command to every node on the ring. The **`ED-FETID`** command generates a REPT-DBCHG message.

Example input/output

The following is an example of the **`ED-FETID`** command:

```
ED-FETID:NENAME-1::123456::RingID:oldtid=NENAME-2,newtid=NENAME-3;
```

```
   NENAME-1 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Automatic ring discovery to be disabled

If the automatic ring discovery is enabled, the following error response is returned:

Error code: `SNVS – Status, Not Valid State`

Validation rule: consistent aid and rid

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: Old TID not in ring map

If the specified Old Target Identifier value is not contained in the list of Ring Node Target Identifiers, the command is denied.

Error code:
`IDNV – Input, Data Not Valid, old TID not in in the list of ring node TIDs`

Validation rule: Local node TID not allowed

If the specified Old Target Identifier is the TID of the local node, the command is denied.

Error code: `IDNV – Input, Data Not Valid, TID of local node not allowed`

Validation rule: No duplicate ring node TIDs

If the specified New Target Identifier is already contained in the list of ring node TIDs and is different from the Old Target Identifier, the command is denied.

Error code: `IDNV – Input, Data Not Valid, duplicate ring node TID`

Related TL1 messages

- "ED-MAP-RING: Edit map ring" (p. 3-436)
- "RTRV-MAP-RING: Retrieve map ring" (p. 3-2532)
- "SET-MAP-RING: Set map ring" (p. 3-3142).

References

GR-199-CORE

ED-FETID notification

Output format

Command syntax: **`ED-FETID`**:*aid*:[*rid*]*:spec_block;*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
2. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: The *rid* is only included in the ED-FETID notification if the *rid* has been specified in the **`ED-FETID`** command.

Spec block parameters

1. **`oldtid`** – Old Target Identifier
   This parameter is the previous TID of a remote node on the ring which may be associated with one or more local cross-connections.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: 1 to 20 characters.
2. **`newtid`** – New Target Identifier
   The newtid parameter is the previous TID of a remote node on the ring which may be associated with one or more local cross-connections.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: 1 to 20 characters.

Example input/output

The following example reports the change of a TID in an NE’s ring map for a 4 fibre MS-SPRing/BLSR protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ED-FETID:1-1-t001:RingID:oldtid=NE_1,newtid=NE_2″
;
```

ED-LONIM: Edit lower-order non-intrusive monitor

The *ED-LONIM* command modifies the properties of a lower-order tributary (LONIM).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax:

`ED-LONIM`-*modifier*:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **`modifier`**
   The modifier specifies the type of the low-order tributary. The purpose is to distinguish between SDH and SONET tributaries.
   Type: Alphanumeric value set.
   Range:
   - LOVC
   - LOVT
   - DS1
2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`aid`** – Access Identifier
   The access identifier specifies a low-order tributary to be operated on.
   Type: AID
   Range: GenLOTribAID, DS1NumAID.
   The following values are not used:
   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID
   - *all* in Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
   - *all* in Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID
   Default: None.
4. **`ctag`** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec Block Parameters

1. **`degthr`** – Signal Degrade Threshold

   This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).

   The parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range: –5, –6, –7, –8, –9; that means, the range is 10^{-9} to 10_{-5}.
   Default: –6.

2. **`dexcthr`** – Excessive Degrade Threshold

   This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
   Default: –3.

3. **`explptrc`** – Expected LO Path Trace Identifier

   This parameter is only significant if *ptrcrfmt* is set to 16. However, setting this parameter while the format is different will not be rejected but stored in the NE and needs not be set again if the format is set later to the 16 byte mode.
   This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding “H”.
   Range: length: 32 hexadecimal; digits: value specific string acc. ITU/ETSI mode 1
   Default: 89000000000000000000000000000000

4. **`feppm`** – Far End Path PM Enable

   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set
Range: DISABLE, ENABLE.
Default: DISABLE.

5. **`fmt`** – Digital Signal Format for DS1
This parameter is used to specify the digital signal format expected to be received from the DS1 at the specified port(s). The TransMUX circuit pack supports DS1 path PM; it does not support DS1 line PM.
The parameter is only supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
   - SF: Super Frame, supports near-end DS1 path and line PM only.
   - ESF: Extended Super Frame, supports near-end and far-end DS1 path and line PM.

   Default: ESF.
6. **`lonimpn`** – LO NIM ASAP Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
The ASAP type depends on the *modifier*:
   - for LOVC/LOVT modifier, an ASAP of type PATH is required
   - for DS1 modifier, an ASAP of type DS1 is required

   Type: Alphanumeric string. A hyphen (" – -") is allowed in the name additionally.
Range: 1 to 24 characters
Default: DEFAULT.
7. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format ptrcrfmt is set to 16 (16-byte specific-string mode).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.
8. **`neppm`** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.

9. **`outpm`** – Outgoing Near End Path PM Enable
The parameter *outpm* specifies whether the performance monitoring for the near end in outgoing direction is enabled or disabled for this tributary.

The parameter is only supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE
Default: DISABLE.

10. **`pmlpfeses`** – PM LO Path Far End SES Threshold
This parameter specifies the Low Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end.
The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range:

The range depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF:

- 1, 2, 3, ..., 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
- 1, 2, 3, ..., 1998, 1999, 2000 (if the entity represents a VC12 tributary)
- 1, 2, 3, ..., 3998, 3999, 4000 (if the entity represents a VT1.5 tributary)

Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary).

11. **`ptca`** – Path TCA Profile Pointer
The parameter *ptca* specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

The TCA Profile type depends on the modifier.

- for LOVC *modifier*, a TCA Profile of type LOVC is required
- for LOVT *modifier*, a TCA Profile of type LOPATH is required
- for DS1 *modifier*, a TCA Profile of type DS1 is required

Type: Alphanumeric string.
Range: 1 … 24 characters.
Default: DEFAULT0.

12. **`pmlpneses`** – PM LO Path Near End SES Threshold
This parameter specifies the Low Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by istd for the related port owning this tributary and the transmission rate of this low-order tributary. For definition of istd refer to RTRV-rr Output Parameters. The transmission rate can only be derived from the parameter inputsig of the related PTF entity.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range:

The range depends on the tributary type in input direction, which can be derived from inputsig of the related PTF:

- 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
- 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
- 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary).
- 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a DS1 tributary)

Default: The default value depends on the tributary type in input direction, which can be derived from inputsig of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary); 1544 (if the entity represents a DS1 tributary).

13. **`ptrcrfmt`** – Path Trace Read Format
This parameter specifies the format of the path trace (J1 or J2 byte) to be received (sink).

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Numeric string.

Range:

- 2 (default), for non-specific repeating byte
- 16, for 16-byte specific string
- 64, for 64-byte string (only applicable if the entity represents a LO-VC3 tributary).

To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.

14. **`tcamd`** – TCA mode

    This parameter is only available for SDH mode and will therefore only be reported if the output parameter *aidtype* is LOVC. The parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable for SDH:

    - TR-only. (also known as the implicit reset method)
    - TR-RTR. (also known as the explicit reset method).

    For SONET the parameter *tcamd* is not supported. SONET ports only support “TR-only” mode. Thus the mode must be internally set to ″“TR-only”.

    The parameter is not supported for the following modifiers:

    - DS1
    - LOVT

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: TR, TRRTR

    Default:

    - SDH mode: TRRTR
    - SONET mode: TR (internal).

    For SONET the parameter *tcamd* is not supported. SONET ports only support “TR-only” mode. Thus the mode must be internally set to ″“TR-only”.

15. **`tmonmd`** – Tributary Monitoring Mode

    This parameter specifies the monitoring mode for the tributary.
    Type: Alphanumeric value set.
    Range: MON, NMON.
    Default: NMON.

Output format

If the network element fully complies with the *ED-LONIM* request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The *ED-LONIM* shall generate a *REPT DBCHG* message. The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed acgLOTrib entity according to the parameters given with this command. The modification is reported by means of an ED-LONIM notification.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts, the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of *tcamd* from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Example input/output

```
ED-LONIM-LOVC:LT-UNITE:1-1-#-#-39-16-1-43:123456:::tmonmd=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the *ED-LONIM* command.

Validation rule: Check *aid* against PTF substructure

If the command is applied with an aid which does not exist according to the input signal substructure of the owning PTF entity then the command is denied.

Error code: `IENE – Input, Entity Not Exists, aid not part of substructure`

Example:

Given the parameter of HPT[1-1-#-#-39-16-1]:inputsig is 23-2112, the following aid will be rejected: 1-1-#-#-39-16-1-43.

Note that the status of the parameter HPT:unequiposig is irrelevant.

The validation rule is not relevant for SONET entities, because SONET PTFs support only one input signal structure, 28 VT1.5/DS1 (2811). SONET PTFs, which do not have an *inputsig* parameter shall implicitly use the signal structure 2811.

Validation rule: Check if parameters are supported for the *modifier*

If the command is applied with spec block parameters which are not supported for the *modifier*, then the command is denied.

Error code: `INUP – Input, Non-null Unimplemented Parameter`

Validation rule: Check *modifier* and *ptrcrfmt* consistency

If the command is applied with the parameter ptrcrfmt which is going to be set to 64 and the addressed entity does not represent a LO VC3 tributary then the command is denied.

Error code:
`IPNC – Input, Parameter Not Consistent, 64 byte path trace not allowed`

64 byte path trace format only allowed for LO-VC3 tributaries. The tributary type the addressed entity represents can only be derived from the parameter inputsig of the related PTF entity.

Validation rule: Check *modifier* and *pmlpneses*/*pmlpnefeses* consistency

If the command is applied for:

- an entity representing a LO-VC3 tributary and one of the parameters *pmlpneses* or *pmlpfeses* is not in the range from 1 to 8000,
- an entity representing a VC12 tributary and one of the parameters *pmlpneses* or *pmlpfeses* is not in the range from 1 to 2000,

- an entity representing a VT1.5 tributary and one of the parameters *pmlpneses* or *pmlpfeses* is not in the range from 1 to 4000,
- an entity representing a DS1 tributary and one of the parameters *pmlpneses* is not in the range from 1 to 8000,

then the command is denied.

Error code:
`IPNC – Input, Parameter Not Consistent, SES threshold out of range for this modifier`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: No LO tributary at HO side of conversion cross-connection

If the LO tributary specified in the command is related to a HO tributary, which is used as HO side of a conversion cross-connection, then the command is rejected.

Error code:
`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Related TL1 messages

- "RTRV-LONIM: Low-order non-intrusive monitor" (p. 3-2486)

ED-LONIM notification

Output format

Command syntax:

`ED-LONIM`-*modifier*:*aid*:*ctag*::*spec_block*;

Output parameters

Command parameters

1. **`modifier`**
   The modifier specifies the type of the low-order tributary. The purpose is to distinguish between: – SDH/SONET tributaries.
   Type: Alphanumeric value set.

   Range:

   - LOVC
   - LOVT
2. **`aid`** – Access Identifier
   The access identifier specifies a low-order tributary to be operated on.
   Type: AID
   Range: GenLOTribAID, DS1NumAID.

   The following values are not used:

   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID
   - *all* in Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
   - *all* in Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID

   Default: None.

Spec Block Parameters

1. **`crslpbkstat`** – Cross-connect Loopback Status
   This parameter shall report the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).

   The parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- YES: for enabled loopback
- NO: for disabled loopback.

2. **`degthr`** – Signal Degrade Threshold
This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).
The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range: –5, –6, –7, –8, –9; that means, the range is 10^{-9} to 10_{-5}.
   Default: –6.

3. **`dexcthr`** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
   Default: –3.

4. **`explptrc`** – Expected LO Path Trace Identifier
This parameter is only significant if *ptrcrfmt* is set to 16. However, setting this parameter while the format is different will not be rejected but stored in the NE and needs not be set again if the format is set later to the 16 byte mode.
This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding “H”.
   Range: length: 32 hexadecimal; digits: value specific string acc. ITU/ETSI mode 1
   Default: 89000000000000000000000000000000

5. **`feppm`** – Far End Path PM Enable
This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set
Range: DISABLE, ENABLE.
Default: DISABLE.

6. **`fmt`** – Digital Signal Format for DS1
This parameter is used to specify the digital signal format expected to be received from the DS1 at the specified port(s). The TransMUX circuit pack supports DS1 path PM; it does not support DS1 line PM.
The parameter is only supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
   - SF: Super Frame, supports near-end DS1 path and line PM only.
   - ESF: Extended Super Frame, supports near-end and far-end DS1 path and line PM.

   Default: ESF.
7. **`lonimpn`** – LO NIM ASAP Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
The ASAP type depends on the *modifier*:
   - for LOVC/LOVT modifier, an ASAP of type PATH is required
   - for DS1 modifier, an ASAP of type DS1 is required

   Type: Alphanumeric string. A hyphen (" – -") is allowed in the name additionally.
Range: 1 to 24 characters
Default: DEFAULT.
8. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format ptrcrfmt is set to 16 (16-byte specific-string mode).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.
9. **`neppm`** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.

10. **`outpm`** – Outgoing Near End Path PM Enable
The parameter *outpm* specifies whether the performance monitoring for the near end in outgoing direction is enabled or disabled for this tributary.
The parameter is only supported for modifier:
    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE
Default: DISABLE.

11. **`pmlpfeses`** – PM LO Path Far End SES Threshold
This parameter specifies the Low Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end.
The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.
The parameter is not supported for modifier:
    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range:
The range depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF:
    - 1, 2, 3, ..., 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
    - 1, 2, 3, ..., 1998, 1999, 2000 (if the entity represents a VC12 tributary)
    - 1, 2, 3, ..., 3998, 3999, 4000 (if the entity represents a VT1.5 tributary)

    Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary).

12. **`pmlpneses`** – PM LO Path Near End SES Threshold
    This parameter specifies the Low Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by istd for the related port owning this tributary and the transmission rate of this low-order tributary. For definition of istd refer to RTRV-rr Output Parameters. The transmission rate can only be derived from the parameter inputsig of the related PTF entity.

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Integer.
    Range:

    The range depends on the tributary type in input direction, which can be derived from inputsig of the related PTF:

    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
    - 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
    - 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary).
    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a DS1 tributary)

    Default: The default value depends on the tributary type in input direction, which can be derived from inputsig of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary); 1544 (if the entity represents a DS1 tributary).
13. **`ptca`** – Path TCA Profile Pointer
    The parameter *ptca* specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    The TCA Profile type depends on the modifier.

    - for LOVC *modifier*, a TCA Profile of type LOVC is required
    - for LOVT *modifier*, a TCA Profile of type LOPATH is required
    - for DS1 *modifier*, a TCA Profile of type DS1 is required

    Type: Alphanumeric string.
    Range: 1 … 24 characters.
    Default: DEFAULT0.
14. **`ptrcrfmt`** – Path Trace Read Format
    This parameter specifies the format of the path trace (J1 or J2 byte) to be received (sink).

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Numeric string.

Range:

- 2 (default), for non-specific repeating byte
- 16, for 16-byte specific string
- 64, for 64-byte string (only applicable if the entity represents a LO-VC3 tributary).

To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.

15. **`tcamd`** – TCA mode

This parameter is only available for SDH mode and will therefore only be reported if the output parameter *aidtype* is LOVC. The parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable for SDH:

- TR-only. (also known as the implicit reset method)
- TR-RTR. (also known as the explicit reset method).

For SONET the parameter *tcamd* is not supported. SONET ports only support “TR-only” mode. Thus the mode must be internally set to ″“TR-only”.

The parameter is not supported for the following modifiers:

- DS1
- LOVT

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: TR, TRRTR

Default:

- SDH mode: TRRTR
- SONET mode: TR (internal).

For SONET the parameter *tcamd* is not supported. SONET ports only support “TR-only” mode. Thus the mode must be internally set to ″“TR-only”.

16. **`tmonmd`** – Tributary Monitoring Mode

This parameter specifies the monitoring mode for the tributary.
Type: Alphanumeric value set.
Range: MON, NMON.
Default: NMON.

Example output

Example of **`ED-LONIM`** notification representing a LO-VC3 tributary:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-LONIM-LOVC:1-1-#-#-39-16-1-43::\
    crslbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lptimmon=DISABLE,lonimpn=DEFAULT,neppm=DISABLE,pmlpfeses=2400,\
    pmlpneses=2400,ptca=DEFAULT0,ptrcrfmt=1,tcamd=TRRTR,tmonmd=NMON″
```

```
;
```

Example of **ED-LONIM** notification representing a VC12 tributary:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-LONIM-LOVC:1-1-#-#-39-16-1-63::\
    crslbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lptimmon=DISABLE,lonimpn=DEFAULT,neppm=DISABLE,pmlpfeses=600,\
    pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=1,tcamd=TRRTR,tmonmd=NMON″
;
```

Example of **ED-LONIM** notification representing a VT1.5 tributary:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-LONIM-LOVT:1-1-#-#-39-16-3-28::\
    crslbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lptimmon=DISABLE,lonimpn=DEFAULT,neppm=DISABLE,pmlpfeses=600,\
    pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=1,tmonmd=NMON″
;
```

Example of **ED-LONIM** notification representing a DS1 tributary:

```
   UNITE-NE 08-03-01 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-50,DATE=08-03-01:ED-LONIM-DS1:\
     1-1-#-#-15-tma1-1-d28::fmt=ESF,outpm=ENABLE,tmonmd=MON″
;
```

ED-LOPTF: Edit lower-order path termination function

The **ED-LOPTF** command modifies the properties of a lower-order path termination tributary (LOPTF).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-LOPTF**-*modifier*:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the low-order tributary.
   Type: Alphanumeric value set.
   Range: TMSVT.
   Default: None.
2. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: TMuxSLONumAID.
   The following values are not used:
   - **all**, **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID
   - **all** for Simple_TMuxSPTNumAID
   - **all** for Simple_TMuxSLONumAID

   No wildcards are allowed.
   Default: None.
4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **degthr** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –5; –6; –7; –8; –9, that means, the range is 10^{-9} to 10^{-5}.
   Default: –6.
2. **dexcthr** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
   Default: –3.
3. **explptrc** – Expected LO Path Trace Identifier
   This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding “H”.
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000
4. **lopn** – LO ASA Profile Name
   This parameter points to a tributary ASAP profile name *pfname* for the designated AID. For *modifier* = TMSVT, an ASAP profile of type *PATHTERM* is required.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 to 24 characters.
   Default: DEFAULT.

5. **`lptimais`** – LO Trace Identifier Mismatch AIS Insertion
   This parameter specifies whether a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - DISABLE
   - ENABLE

   Default: The default value depends on interface standard:
   - SONET: DISABLE.
6. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
   This parameter specifies the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - DISABLE
   - ENABLE

   Default: DISABLE
7. **`ptrcrfmt`** – Path Trace Read Format
   The parameter *ptrcrfmt* specifies the format of the path trace (J1 or J2 byte) to be received (sink).
   The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Numeric string.
   Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string
   - 64: for 64-byte string.

   To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.
   Default: 2.
8. **`ptrcwfmt`** – Path Trace Write Format
   This parameter specifies the format of the path trace (J2 byte) to be sent (source).
   Type: Numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16-byte specific string

Default: 2.

9. **`tmonmd`** – Tributary Monitoring Mode
   This parameter specifies the monitoring mode for the tributary.
   Type: Alphanumeric value set.
   Range:
   - MON
   - NMON

   Default: NMON.
10. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J2) to be sent (source). For the 16-byte path trace format, the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    Type: Hexadecimal number without a preceding "H".
    Range: The range depends on the parameter *ptrcwfmt* specified in **`RTRV-PTF`** output parameters
    - length: 32 hexadecimal digits.
    - value: specific string acc. ITU/ETSI mode 1.

    Default: 00.

Output format

If the **`ED-LOPTF`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`ED-LOPTF`** shall generate a **`REPT DBCHG`** message. The command shall modify the attributes (listed in the spec_block parameters part of this command) of the addressed LO Path Termination entitity according to the parameters given with this command. The modification is reported by means of an **`ED-LOPTF`** notification.

Example input/output

```
ED-LOPTF-TMSVT:LT-UNITE:1-1-#-#-39-tms16-1-28:123456:::tmonmd=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-LOPTF** command.

Validation rule: Check *modifier* and *ptrcrfmt* consistency

If the command is applied with the parameter *ptrcrfmt* which is going to be set to 64, then the command is denied.

Error code:

`IPNC – Input, Parameter Not Consistent, 64 byte path trace not allowed`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Related TL1 messages

- "RTRV-LOPTF: Retrieve low-order path termination function" (p. 3-2478)

ED-LOPTF notification

Output format

Command syntax: **ED-LOPTF**-*modifier*:*aid*::*spec_block*;

Output parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the low-order tributary.
   Type: Alphanumeric value set.
   Range: TMSVT.
   Default: None.
2. **aid** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: TMuxSLONumAID.
   The following values are not used:
   - **all**, **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID
   - **all** for Simple_TMuxSPTNumAID
   - **all** for Simple_TMuxSLONumAID

   No wildcards are allowed.
   Default: None.

Spec block parameters

1. **degthr** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –5; –6; –7; –8; –9, that means, the range is 10^{-9} to 10^{-5}.
   Default: –6.
2. **dexcthr** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
Default: –3.

3. **`explptrc`** – Expected LO Path Trace Identifier
   This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding "H".
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000
4. **`lopn`** – LO ASA Profile Name
   This parameter points to a tributary ASAP profile name *pfname* for the designated AID. For *modifier* = TMSVT, an ASAP profile of type *PATHTERM* is required.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: 1 to 24 characters.
   Default: DEFAULT.
5. **`lptimais`** – LO Trace Identifier Mismatch AIS Insertion
   This parameter specifies whether a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
   Type: Alphanumeric value set.

   Range:

   - DISABLE
   - ENABLE

   Default: The default value depends on interface standard:

   - SONET: DISABLE.
6. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
   This parameter specifies the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- DISABLE
- ENABLE

Default: DISABLE

7. **`ptrcrfmt`** – Path Trace Read Format
The parameter *ptrcrfmt* specifies the format of the path trace (J1 or J2 byte) to be received (sink).
The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Numeric string.
Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string
   - 64: for 64-byte string.

   To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.
Default: 2.
8. **`ptrcwfmt`** – Path Trace Write Format
This parameter specifies the format of the path trace (J2 byte) to be sent (source).
Type: Numeric string.
Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string

   Default: 2.
9. **`tmonmd`** – Tributary Monitoring Mode
This parameter specifies the monitoring mode for the tributary.
Type: Alphanumeric value set.
Range:
   - MON
   - NMON

   Default: NMON.
10. **`trc`** – Outgoing Path Trace
This parameter specifies the path trace (J2) to be sent (source). For the 16-byte path trace format, the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
Type: Hexadecimal number without a preceding “H”.

Range: The range depends on the parameter *ptrcwfmt* specified in **RTRV-PTF** output parameters

- length: 32 hexadecimal digits.
- value: specific string acc. ITU/ETSI mode 1.

Default: 00.

Example input/output

Example of ED-LOPTF notification representing a VT1.5 tributary:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-LOPTF-TMSVT:1-1-#-#-39-tms16-1-28::\
    explptrc=89000000000000000000000000000000,\
    lptimmon=DISABLE,lopn=DEFAULT,\
    ptrcrfmt=1,tmonmd=NMON″
;
```

ED-MAP-RING: Edit map ring

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No.

Input format

Command syntax: **`ED-MAP-RING`**:*tid*:[*aid*]:*ctag*::[*rid*],*spec_block;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **`ctag`** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: If *aid* is specified, then *rid* can be omitted. If *rid* is omitted, the *aid* parameter is used to select a ring. The values of *aid* and *rid* must not both be omitted.

Spec block parameters

1. **`localnid`** – Local Node ID
   This parameter specifies the local node ID of the NE receiving the retrieve request (i.e. the node which has been addressed within RTRV-MAP-Ring command) as used by the MS-SPRing/BLSR APS protocol.
   Type: Numeric string.
   Range: 0 .. 15
   Option: The operator is responsible of the uniqueness of the value in the ring.
2. **`ringdir`** – Ring Direction
   This parameter indicates whether the local node's east or west provisioned or hardwired direction port is connected to the next node in the ring map.
   Type: Alphanumeric string.
   Range: E.
3. **`rnids`** – Ring Node Identifiers
   This is an ampersand-separated list with a maximum of 16 entries, where each entry is an integer between 0 and 15. The first entry is the lowest numbered node ID on the ring. Each subsequent value is the node IF of the next node on the ring. Each permitted value may only be used once in the list. The number of values defines the number of nodes on the ring. If the number of nodes specified is smaller than previously specified, those unspecified entries in the ring map are set to the default value.
   Type: string.
   Range: All permutations of a subset of {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15} separated by a '&'.
4. **`rtids`** – Ring Node Target Identifiers
   This is an ampersand-separated list with a maximum of 16 entries, where each entry is a valid TID. Each entry in the list is the TID corresponding to the entry in that position of the *rnids* list. Similarly, if the size of the ring map decreases from the previous ED-MAP-RING command, those unspecified TID entries in the ring map are set to the default value. The number of values for *rnids* and *rtids* must be equal.
   Option: If *aid* is specified, then *rid* can be omitted.
   Type: string.
   Range: 16 ampersand-separated TIDs.
   Default: None (empty).

Note:

When different numbers of ring node identifiers (*rnids*) and ring node target identifiers (*rtids*) are entered, the command is denied. However, the related error message is not precise enough, but it simply states: `/* Input, Data Not Valid */`. The correct error message should be:

```
/* Input, Data Not Valid, different number of values in rnids and rtids
*/
```

Output format

After receiving the **`ED-MAP-RING`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

1. **`sid`** – Source Identifier
   This is the currently active name of the network element to which the command is addressed. The tid is a case insensitive character string.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: 1 to 20 characters.
2. **`date`** – Date
   Date when output message is generated. This has the format YY-MM-DD (year-month-day)
   Type: value set.
   Range: for YY 00 to 99, for MM 01 to 12, for DD 01 to 31
   TL1 messages use YY-MM-DD for the format of a date, meaning that the year only has two digits. This is consistent with Telcordia requirements. Internally, however, the system does use a 4-digit year. When the year appears in a response, the internal 4-digit year is truncated to two digits. When the year appears as an input parameter, for example in the **`ED-DAT`** command, years in the range 90-99 will represent 1990-1999, and years in the range 00-37 will represent 2000-2037.
   Default:00-01-01
3. **`time`** – Time
   Time when output message is generated. This has the format HH:MM:SS (hours:minutes:seconds).
   Type: Value set.
   Range: for HH 00 to 23, for MM 00 to 59, for SS 00 to 59.
   Default: 00:00:00
4. **`M`** – Message
   M indicates that the output message is generated in response to a manual command.
   Type: Value set.
   Range: M

5. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: Non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
6. **IP** – In Progress
   IP is sent in response to all commands with the exception of commands that result in error code output.
   Type: Value set.
   Range: IP
7. **RL** – Repeat Later
   The requested command cannot be executed due to unavailable system resources. The command may be entered again later.
   Type: Value set.
   Range: RL
8. **COMPLD** – Completion Code
   It indicates that the command has been completed.
   COMPLD for successful command execution: For a command that is decomposed into a set of atomic commands a COMPLD completion code means that the command was successful for all atomic commands (e.g. AID ranges, group alias).
   COMPLD for no changes: If a parameter has a current value, and a provisioning command attempts to reprovision the parameter with a value that is the same as its current value, the command will return a COMPLD completion code.
   Type: Value set.
   Range: COMPLD.

Detailed behavior description

For open rings, the output shall start at the end with the lower nid value and shall continue to the other end. For closed rings, the output shall start at the node with the lowest *nid* value and shall continue in the direction of the first node’s east neighbor. The **ED-MAP-RING** command is also denied if the list does not contain the local node , if the list does not contain at least 2 nodes, or if duplicate nid values appear in the list.

Example input/output

The following is an example of the **ED-MAP-RING** command. Manually, node numbers 3 for NENAME-1, 14 fr NENAME-2 and 8 for NENAME-3 are assigned:

```
ED-MAP-RING:NENAME-1::123456::RingID:rnids=3&14&8,rtids=NENAME-1&NENAME-2&NENAME-3;
   NENAME-1 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: consistent aid and rid

If both an *aid* and an *rid* are specified as input parameters and they are inconsistent, then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: No duplicate Ring Node Identifiers

If the list of ring node identifiers specified in the parameter rnids contains duplicate values, the command is denied.

Error code: `IDNV – Input, Data Not Valid, duplicate ring node identifiers`

Validation rule: No duplicate Ring Node Target Identifiers

If the list of ring node target identifiers specified in the parameter rtids contains duplicate values, the command is denied.

Error code:
`IDNV – Input, Data Not Valid, duplicate ring node target identifiers`

Validation rule: Consistent *rnids* and *rtids* lists

The lists of ring node identifiers and ring node target identifiers specified in the parameters *rnids* and *rtids* shall contain the same number of values, otherwise the command is denied.

Error code:
`IDNV – Input, Data Not Valid, different number of values in rnids and rtids`

Validation rule: Illegal local node ID

If the local node identifier specified with the parmater localnid is not contained in the list of ring node identifers, or if the local node identifier has not been specified, the command is denied.

Error code:
`SNVS – Status, Not in Valid State, illegal local node identifier`

Validation rule: Less than 2 nodes specified

If the list of ring node identifiers contains less than 2 values, the command is denied.

Error code: `INUP – Input, Non-null Unimplemented Parameter`

Related TL1 messages

- "ED-FETID: Edit far end target identifier" (p. 3-405)§
- "ED-PROTN-GRP: Edit protection group" (p. 3-803)
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
- "RTRV-MAP-RING: Retrieve map ring" (p. 3-2532)
- "SET-MAP-RING: Set map ring" (p. 3-3142).

References

GR-199-CORE

ED-NE: Edit NE

The **ED-NE** command is used to activate or de-activate DCN functions associated with the NE at the system level, e.g. the IP tunneling function or the TL1 Translation Device (T-TD) function.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-NE**:*tid*::*ctag*:::*spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
Description: Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
Description: Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
Type: CTAG.

Spec block parameters

1. **ftamftpgw** – File Transfer Translation Device (FT-TD) function
This parameter is used to enable or disable the File Transfer Translation Device (FT-TD) function.
Type: Value set.
Range: ENABLE, DISABLE (default).
2. **tl1tcpipgw** – TL1 Translation Device (T-TD) function
This parameter is used to enable or disable the TL1 Translation Device (T-TD) function.
Type: Value set.
Range: ENABLE, DISABLE (default).

3. **dcnpn** – DCN ASAP Profile Name
   This parameter specifies the name of the current ASAP profile used for DCN alarms.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
4. **scnpn** – SCN ASAP Profile Name
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.

Output format

If the **ED-NE** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command activates or de-activate the DCN function associated with the parameters specified in the *spec_block* section.

Disabling *ftamftpgw*

When the parameter ftamftpgw is set to DISABLE existing file transfers via FT-TD shall complete normally, but no new file transfer via FT-TD is allowed.

Example input/output

The following example shows the result of an ED-NE command, where IP tunneling is enabled:

```
ED-NE:LT-UNITE::123456:::scnpn=PROFILE2;

   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the *RTRV-NE* command.

Validation rule: Check for correct IP configuration

The request to ENABLE *ftamftpgw*, *tl1tcpipgw* or *tunstat* is only allowed if there is already a valid IP configuration (IP address of LAN1, subnet mask of LAN1, default router) provisioned for the system.

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

For the specific checks refer to

- ENT-FECOM-LAN, parameter ip_lanstat
- ENT-FECOM-LAN, parameter ip_snm
- ENT-ULS, parameter ip_drad.

Validation rule: Check for correct TARP configuration

The request to ENABLE *tl1tcpipgw* is only allowed if TARP originator, propagator and responder roles are enabled for node 1.

Error code: `SNVS – Status, Not in Valid State, TARP not enabled`

Validation rule: Modification of SCN related parameters only while ONNS is enabled

The system shall not allow to modify the SCN related parameters *lsprepholdoff* and *scnpn* while ONNS is disabled (system parameter xcsetup=TRADITIONAL).

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Related TL1 messages

- “RTRV-NE: Retrieve NE” (p. 3-2549)
- “RTRV-FECOM-LAN: Retrieve LAN channel related parameters” (p. 3-2413)
- “ENT-FECOM-LAN: Enter LAN related parameters” (p. 3-1581)

ED-NE notification

Output format

The **ED-NE** notification has the following specific syntax:

ED-NE:::*spec_block*

The output parameters *sid*, *date*, *time*, and *atag* are described in the **REPT DBCHG** notification.

Output parameters

1. **ftamftpgw** – File Transfer Translation Device (FT-TD) function
   This parameter is used to enable or disable the File Transfer Translation Device (FT-TD) function.
   Type: Value set.
   Range: ENABLE, DISABLE (default).
2. **tl1tcpipgw** – TL1 Translation Device (T-TD) function
   This parameter is used to enable or disable the TL1 Translation Device (T-TD) function.
   Type: Value set.
   Range: ENABLE, DISABLE (default).
3. **dcnpn** – DCN ASAP Profile Name
   This parameter specifies the name of the current ASAP profile used for DCN alarms.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
4. **scnpn** – SCN ASAP Profile Name
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.

Example output

The following is an example ED-NE notification in case the *tunstat* function has been enabled:

```
   LT-UNITE 06-06-30 08:00:00
A  000 REPT DBCHG
   ″TIME=08-00-00,DATE=06-06-30:ED-NE:::scnpn=PROFILE2″

;
```

ED-NE-SECU: Edit NE security

The **ED-NE-SECU** command can be initiated by a user to modify global security parameters in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No

Input format

Command syntax: **ED-NE-SECU**:*tid*::*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Spec block parameters

1. **age** – Lockout Aging Period
   The time interval during which the number of invalid login attempts are counted. When *age* expires after the last invalid login attempt, the counter of invalid login attempts by that *uid* is reset to zero.
   Type: Numeric string.
   Range: 1 – 999 [minute]
   Default: 60
2. **intrvl** – Lockout Period
   The time during which login attempts using that *uid* will not be allowed. This occurs when the number of consecutive invalid login attempts by a *uid* reaches *thrshld*. When *intrvl* expires, login attempts will again be allowed from that *uid*. If this parameter is set to 0 (zero), the lockout feature is disabled.
   Type: Numeric string.
   Range: 0 – 99 [minute]
   Default: 10

3. **`thrshld`** – Lockout Threshold
   The maximum number of consecutive invalid login attempts that are allowed by a *uid*. If the count of invalid login attempts is equal to the threshold, the *uid* is locked out for *intrvl* period of time. If a login attempt is successful, the count of *invalid* login attempts by that *uid* is reset to zero.
   Type: Numeric string.
   Range: 2 – 99
   Default: 5
4. **`usrage`** – User ID Aging Period
   If a *uid* has never been used to perform a successful login during a specified time interval it is disabled. A value of 0 disables user ID aging.
   The *usrage* does not apply to the super-users. Super-users will be never disabled.
   Type: Numeric string.
   Range: 0 – 999 [days]
   Default: 10

Output format

After receiving the **`ED-NE-SECU`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The NE returns the successful completion response even if none of the parameter values are modified using the **`ED-NE-SECU`** command.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **`RTRV-HDR`** command.

Detailed behavior description

This command updates the NE security administration values of *thrshld* (Lockout Threshold), *intrvl* (Lockout Period), *age* (Lockout Aging Period), *usrage* (User ID Aging Period) according the values in the parameters .

This update is reported via the corresponding ED-NE-SECU notification – see "ED-NE-SECU notification" (p. 3-449).

Example input/output

The following example shows the successful completion of the **`ED-NE-SECU`** command by the NE:

`ED-NE-SECU:LT-UNITE-123::123456:::thrshld=5,intrvl=2,age=50,usrage=200;`

```
   LT-WBM-789 00-01-01 08:00:00
M  123456 COMPLD
```

;

Error responses

The general error responses defined in “RTRV-HDR: Retrieve header” (p. 3-2417) are also valid for this command.

Related TL1 messages

- “ACT-USER: Activate user” (p. 3-24)
- “RTRV-NE-SECU: Retrieve NE security” (p. 3-2552)

ED-NE-SECU notification

Output format

The **ED-NE-SECU** notification has the following specific syntax:

ED-NE-SECU:::*spec_block*

Output parameters

Spec block parameters

1. **age** – Lockout Aging Period
   The time interval during which the number of invalid login attempts are counted. When age expires after the last invalid login attempt, the counter of invalid login attempts by that uid is reset to zero.
   Type: Numeric string.
   Range: 1 – 999 [minute]
   Default: 60
2. **intrvl** – Lockout Period
   The time during which login attempts using that uid will not be allowed. This occurs when the number of consecutive invalid login attempts by a uid reaches thrshld. When intrvl expires, login attempts will again be allowed from that uid. If this parameter is set to 0 (zero), the lockout feature is disabled.
   Type: Numeric string.
   Range: 0 – 99 [minute]
   Default: 10
3. **thrshld** – Lockout Threshold
   The maximum number of consecutive invalid login attempts that are allowed by a uid. If the count of invalid login attempts is equal to the threshold, the uid is locked out for intrvl period of time. If a login attempt is successful, the count of invalid login attempts by that uid is reset to zero.
   Type: Numeric string.
   Range: 2 – 99
   Default: 5
4. **usrage** – User ID Aging Period
   Type: Numeric string.
   Range: 0 – 999 [days], default: 10

Example output

The example shows a normal notification after an ED-NE-SECU command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-NE-SECU:::thrshld=5,intrvl=2,age=50,usrage=200″
;
```

ED-NNCNFG: Modify ONNS system configuration

Overview

The **ED-NNCNFG** command is used to modify ONNS system level parameters.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P4

Abortable: No

Input format

Command syntax: **ED-NNCNFG:***srcTid:srcAid:ctag::specBlock;*

Input parameters

Command parameters

1. **srcTid** – Source Target Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set.
2. **srcAid**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **ctag** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

Spec block parameters

1. **`AdminCostEscalation`**
   Determines whether a cost escalation algorithm is applied or not. This algorithm raises the cost of a link bundle to reduce the attractiveness when the link bundle fills up.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE (default).
2. **`DefaultRstrPriority`** – Default Restoration Priority
   This parameter determines the connection restoration priority default to be used if a value is not specified in the **`ENT-PATH`** command. Lower values denote higher priority.
   Note: Though the parameter *DefaultRstrPriority* can be set via ED-NNCNFG, it is for future use only, i.e. an actual value different from 0 will not be considered on connection setup. If no restoration priority is given via the **`ENT-PATH`** command, the priorty of the corresponding connection is always 0, no matter of the actual value of *DefaultRstrPriority*.
   Type: Integer.
   Range: 0 – 7).
   Default: 0.
3. **`NNFunctionUpgradeState`**
   After a software upgrade, new ONNS functionality must be explicitly enabled. This parameter indicates whether this has been done or not.
   Type: Alphanumeric value set.
   Range:
   - ENABLE: New ONNS functionality already enabled.

   Default: ENABLE.
4. **`TCAThresholdHigh`**
   Upper threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status HIGH for this link bundle
   Type: Integer
   Range: 10 – 100 %
   Default: 70 %

5. **`TCAThresholdLow`**
   Lower threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status LOW or HIGH for this link bundle.
   Type: Integer
   Range: 10 – 100%
   Default: 40%
6. **`WaitToRevertTimer`**
   Determines how long an original path must be steadily OK before it is made active again through tear-down of the restoration path. Applies only to auto-reroute revertive connections. This parameter shall apply to M:N protected connections as well.
   Type: Integer
   Range: 60 – 1800 seconds
   Default: 300 seconds.

Note: When parameters *WaitToRevertTimer* or *DefaultRstrPriority* have been changed via the **`ED-NNCNFG`** command, such changes will not be confirmed by the system with a REPT-NNCHG notification. Execute RTRV-NNCHG to get confirmation about changed values from the system.

Output format

If the **`ED-NNCNFG`** request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″specBlock″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completionCode* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

Spec block parameters

This block contains the descriptions and allowed values for the reported output parameters.

1. **`AdminCostEscalation`**
   Determines whether a cost escalation algorithm is applied or not. This algorithm raises the cost of a link bundle to reduce the attractiveness when the link bundle fills up.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE (default).
2. **`DefaultRstrPriority`** – Default Restoration Priority
   Determines the connection restoration priority default to be used if a value is not specified in the **`ENT-PATH`** command. Lower values denote higher priority.
   Note: Though the parameter *DefaultRstrPriority* can be set via ED-NNCNFG, it is for future use only, i.e. an actual value different from 0 will not be considered on connection setup. If no restoration priority is given via the **`ENT-PATH`** command, the priorty of the corresponding connection is always 0, no matter of the actual value of *DefaultRstrPriority*.
   Type: Integer
   Range: 0 – 7 (default = 0, highest priority).
3. **`NNFunctionUpgradeState`**
   After a software upgrade, new ONNS functionality must be explicitly enabled. This parameter indicates whether this has been done or not.
   Type: Alphanumeric value set.

   Range:

   - ENABLE: New ONNS functionality already enabled.

   Default: ENABLE.
4. **`TCAThresholdHigh`**
   Upper threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status HIGH for this link bundle
   Type: Integer
   Range: 10 – 100 %
   Default: 70 %

5. **`TCAThresholdLow`**
   Lower threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status LOW or HIGH for this link bundle.
   Type: Integer
   Range: 10 – 100 %
   Default: 40 %

6. **`WaitToRevertTimer`**
   Determines how long an original path must be steadily OK before it is made active again through tear-down of the restoration path. Applies to auto-reroute revertive connections. This parameter shall apply to M:N protected connections as well.
   Type: Integer
   Range: 60 – 1800 seconds
   Default: 300 seconds

Example input/output

The following example demonstrates the usage of the **`ED-NNCNFG`** command for provisioning the system configuration of the wait to revert timer to 60 seconds:

```
ED-NNCNFG:LT-ONNS-A::123456::waitToRevertTimer=60;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”WaitToRevertTimer=60”
;
```

The following example demonstrates the usage of the **`ED-NNCNFG`** command for provisioning the system configuration of the default connection restoration priority to 5:

```
ED-NNCNFG:LT-ONNS-A::123456::defaultRstrPriority=5;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”DefaultRstrPriority=5”
;
```

The following example demonstrates the usage of the **`ED-NNCNFG`** command for provisioning the system configuration of the lower TCA threshold to 42%:

```
ED-NNCNFG:LT-ONNS-A::123456::TCAThresholdLow=42;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”TCAThresholdLow=42”
;
```

The following example demonstrates the usage of the **ED-NNCNFG** command for provisioning the system configuration of the upper TCA threshold to 69%:

```
ED-NNCNFG:LT-ONNS-A::123456::TCAThresholdHigh=69;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”TCAThresholdHigh=69”
;
```

The following example demonstrates the usage of the **ED-NNCNFG** command for disabling the cost escalation algorithm:

```
ED-NNCNFG:LT-ONNS-A::123456::AdminCostEscalation=DISABLE;

   LT-ONNS-A 2003-08-01 09:00:35
M  123456 COMPLD
   ”AdminCostEscalation=DISABLE”
;
```

The following example demonstrates the usage of the **ED-NNCNFG** command for the provisioning of multiple ONNS system configuration parameters:

```
ED-NNCNFG:LT-ONNS-A::123456::waitToRevertTimer=120,defaultRstrPriority=4;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”WaitToRevertTimer=120,
   DefaultRstrPriority=4”
;
```

The following example demonstrates the usage of the **ED-NNCNFG** command to enable new ONNS functionality after a SW upgrade:

```
ED-NNCNFG:LT-ONNS-A::123456::NNFunctionUpgradeState=ENABLE;

   LT-ONNS-A 2002-08-01 08:00:24
M  123456 COMPLD
   ”NNFunctionUpgradeState=ENABLE”
;
```

Error responses

Command with an invalid Spec bloc parameter

If the NE receives an ED-NNCNFG command with an invalid specific block input named parameter, then one of the following error responses is returned:

Error code:

`E030 – Syntax, Parameter Data Invalid, invalid_parameter=fieldName`

Error code:

`E246 – Syntax, Parameter Name Invalid, invalid_parameter=fieldName`

TCA related error messages

If the NE receives an ED-NNCNFG command with a *TCAThresholdHigh* value which is out of range, then the following error response is returned:

Error code:

`E420 – Input, Parameter out of Range,invalid_parameter=TCAThresholdHigh`

If the NE receives an ED-NNCNFG command with a *TCAThresholdLow* or *TCAThresholdHigh* which does not fulfill the condition *TCAThresholdLow* <= *TCAThresholdHigh*, the following error response is returned:

Error code: `E422 – Input, TCAThresholdLow Greater than TCAThresholdHigh`

If both parameters are changed with the same command, the validation will be done among these two parameters, otherwise the new value will be compared against the value of the existing second TCA threshold value.

ED-NNCRS: Edit NN cross-connection

The **ED-NNCRS**-modifier command can be initiated by a user to set the cross-connection class parameter (*xconcls*) to TRADITIONAL. The command is available for privileged users as a precondition to delete a cross-connection with *xconcls* = NN via DLT-CRS.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P5

Priority: 1

Abortable: No

Input format

Command syntax: **ED-NNCRS**-*modifier*:*tid*:*in_aid*,*out_aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **ED-NNCRS** command acts on.
   For detailed description please refer to the **RTRV-CRS** command.
   Type: Alphanumeric value set.
   Range: VC3, VC4, VC44C, VC416C, VC464C, STS1, STS3, STS12, STS48, STS192.
   No **all** value is supported.
2. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and the description of this parameter.
3. **in_aid** – Input Access Identifier
   This parameter specifies the source of the cross-connection. For a detailed description refer to the *in_aid* output parameter in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
   Type: GenTribAID.
   The GenTribAID includes the TMuxAPTNumAID, which identifies the HO path terminations of asynchronous TransMUX ports. Because the **ED-NNCRS** command can only be used for cross-connections with class NN, and TransMUX ports cannot be used for ONNS, the TMuxAPTNumAID will never be applicable for this command.
   Range: No use of the **all** or **usall** or **vall** values within the *in_aid* is allowed.

4. **`out_aid`** – Output Access Identifier
   This parameter specifies the sink of the cross-connection. For a detailed description refer to the *out_aid* output parameter in "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).
   Type: GenTribAID.
   The GenTribAID includes the TMuxAPTNumAID, which identifies the HO path terminations of asynchronous TransMUX ports. Because the **`ED-NNCRS`** command can only be used for cross-connections with class NN, and TransMUX ports cannot be used for ONNS, the TMuxAPTNumAID will never be applicable for this command.
   Range: No use of the **`all`** or **`usall`** or **`vall`** values within the *in_aid* is allowed..
5. **`ctag`** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **`xconcls`** – cross-connection class
   This parameter identifies, whether the cross-connection has been switched explicitly on the traditional way via an ENT-CRS command or implicitly via setting up an ONNS path. It can be set to TRADITIONAL via this command.
   Type: Alphanumeric value set.
   Range: TRADITIONAL

Output format

If the **`ED-NNCRS`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of the "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The *xconcls* parameter of a cross-connection is set to TRADITIONAL via the **`ED-NNCRS`** command.

The change is reported via an ED-CRS (not ED-NNCRS!) notification.

Path-protected cross-connections

If the **`ED-NNCRS`** command is applied on one leg of a path-protected cross-connection, then the cross-connection class of both legs of the path-protected cross-connection is changed.

2-way cross-connections

If the **`ED-NNCRS`** command is applied on one direction of a 2-way cross-connection, then only the cross-connection class of the specified direction is changed. The class of the other direction shall remain unchanged.

Example input/output

```
ED-NNCRS-VC4:UNITE-NE:1-1-#-#-10-1-4,1-1-#-#-20-1-7:123456:::xconcls=TRADITIONAL;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Entities (tributaries) must exist

Cross-connection command processing verifies that the logical tributaries specified in the command do exist.

Error code:

`IENE – Input Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Entity (cross-connection) must exist

Cross-connection command processing verifies that the specified cross-connection exists.

Error code:

`IENE – Input Entity Not Exists, Specified crossconnection does not exist`

Validation rule: Specified rate consistent with the cross-connecton rate

Cross-connection command processing verifies that the specified rate (*modifier* parameter) is compliant with the rate of the existing cross-connection.

Error code:

`IPNC – Input Parameter Not Consistent, Rate inconsistent with cross-connection rate`

Validation rule: Cross-connection must have *xconcls* = NN

Cross- connection command processing shall verify that the specified cross-connection has set *xconcls* = NN.

Error code:

`SNVS – Status, Not in Valid State, Cross-connection not of class NN`

Validation rule: Cross-connection not involved in Test Access Session

Cross-connection command processing shall verify that the specified cross- connection is not involved in a Test Access Session.

Error code:

`SNVS – Status, Not in Valid State, Cross-connection involved in Test Access Session`

Related TL1 messages

- “DLT-CRS: Delete cross-connection” (p. 3-145)
- “ENT-CRS: Enter cross-connection” (p. 3-1464)
- “ENT-ROLL: Enter roll” (p. 3-1796)
- “RTRV-CRS: Retrieve cross-connection” (p. 3-2306)

ED-OC3: Edit OC3 port

The **ED-OC3** command is used to provision all parameters related to optical OC3 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-OC3**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC3 port.
   Type: AID.
   Range: OC3 port AID
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.

3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

6. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
7. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
8. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is setable and retrievable as in fixed mode but does not influence the port behavior.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
   Default: 31.
   Example: An O-3 signal might be represented by 31 which implies 3 STS1's or 13 which implies 1 STS3C.
9. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SONET, SDH.
   Default: As specified via *istddflton* system level, see definition in RTRV-EQPT (system).
10. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

11. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
12. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
13. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 or STS3 (default)

14. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default) or STS3.
15. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
16. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
17. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If

ndetmd = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

18. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

19. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

20. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

21. **`nelpm`** – Near-End Line PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).

22. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.
    Range:
    - IPV4 (default)
    - IPV6
    - NSAP

23. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

24. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-1/OC-3: 1 – 192000.

Default: The default values differ for SDH and SONET:

- STM1: 28800
- OC3: 154.

25. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM1: 28800
    - OC3: 154.
26. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
27. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM1: 2400
    - OC3: 155.
28. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

29. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

30. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.

31. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

32. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

33. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
34. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: *""*
35. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.
36. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

37. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
38. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
39. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
40. **`sfthr`** – Signal Failure Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –3; –4; –5.
    Default: –3.
41. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
42. **`sonet`** – SONET ASAP
    SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see "RTRV-STM4: Retrieve STM4 port" (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
    Type: Alphanumeric string.
    Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
    Default: DEFAULT.

43. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
44. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
45. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
46. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

47. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
48. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400) command.
49. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port’s bandwidth. It determines the rate(s) of the “unequipped” signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
Default: 3 1 = all STS1.

50. **`z0outmode`** – Mode of outgoing Z0 byte insertion
    The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
    Type: Alphanumeric value set.
    Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Output format

If the ED-OC3 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OC3 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SONET to SDH has the following impact related to Fault Management:

- all SONET alarms related to that port are cleared
- the corresponding SDH alarms are raised.

Switching of tributary mode

When changing the parameter *trbmd* – tributary mode from FIXED to ADAPTIVE the parameter *crstat* may change its value, too, depending on the current input signal structure. This change is notified using the related ED-OC3 notification.

When changing the parameter *trbmd* – tributary mode from ADAPTIVE to FIXED the parameter *crstat* may change its value, too, depending on the current setting of parameter *inputsig*. This change is notified using the related ED-OC3 notification.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

The switching of *istd* should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *degthr* is set to the default value for an SDH port
- *dexcthr* is set to the default value for an SDH port
- *felpm* is mapped to *femspm* (no changes)
- *inputsig* is set to the default value

- *istd* is set to SDH
- *nelpm* is mapped to *nemspm* (no changes)
- *rstimca* is set to the default value for a SDH port
- *rstimmon* is set to the default value for a SDH port
- *sdsfmode* is set to the default value for an SDH port
- *spm* is mapped to *rspm* (no changes)
- *sonet* is mapped to *sdhpn* (no changes)
- *trbmd* is set to the default value for an SDH port
- *unequiposig* is set to its default value.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *rsmstca* is set to the default value for an SDH port
- *tcamd* is set to the default value for a SDH port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmmsfeses* is set to the default value for a SDH port
- *pmmsneses* is set to the default value for a SDH port
- *pmrsneses* is set to the default value for a SDH port.

Switching of interface standard IV:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- **`minNNXCRate`** is set to the default value for a SDH port
- **`maxNNXCRate`** is set to the default value for a SDH port

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value for a SDH tributary
- *brstthr* is set to the default value for a SDH tributary
- *pse* (PDI-P parameter) is set to the default value for a SDH tributary
- *sdsfmode* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *degthr* is set to the default value for an SDH tributary
- *dexcthr* is set to the default value for an SDH tributary
- *ptca* is set to the default value for a SDH tributary
- *tcamd* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmhpneses* is set to the default value for an SDH tributary
- *pmhpfeses* is set to the default value for an SDH tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-OCn command all resulting parameter changes are reported by ED-STMn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-OCn command all resulting tributary parameter changes are reported by ED-VC3 notifications.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual ***portClass***

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when ***portClass*** is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when ***portClass*** is set to INNI and ***ndetmd*** is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when ***portClass*** is set to INNI and ***ndetmd*** is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters, values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*

- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-OC3:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=31,UNEQUIPOSIG=31;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-OC3:1-1-#-#-1-1::inputsig=31,unequiposig=31:"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-OC3 command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no test access session for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: no protection for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no path protection for *trbmd* change to ADAPTIVE

Check that *trbmd* is not changed to ADAPTIVE when tributaries of this port are used as input tributaries in a path protection group.

Error code:

`SNVS – Status, Not in Valid State, a path protection group is existing`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, 'ALL' qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

```
SNVS – Status, Not in Valid State, only allowed for INNI port in
lockout state
```

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

```
SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI
parameters
```

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, ndetmd must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

```
IDNV – Input, Data Not Valid, zero values can only be set for all
neighbor information consistently
```

ONNS EDGE PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 4f-BLSR group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 4f-BLSR protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following preconditions must be fulfilled:

- No cross-connections of *xconcls* = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: No modification of *trbmd* to ADAPTIVE at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from FIXED to ADAPTIVE for ports with *portClass* EDGE, while NN cross-connections exist at the port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* to FIXED at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from ADAPTIVE to FIXED for ports with *portClass* EDGE, while NN cross-connections or traditional cross-connections exist at the port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* EDGEwhile *portControl* is not LOCKOUT

It is possible to modify the *trbmd* for an EDGE port only

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to EDGE

Error code:
```
SNVS – Status, Not in Valid State, only allowed for EDGE port in
lockout sate
```

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters if there still any cross-connections for the EDGE port exist which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

```
SNVS – Status, Not in Valid State, Port must not be a 1+1 protection
port
```

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI if the port is the **protection port** of a 1:1 MSP/APS group.

Error code:

```
SNVS – Status, Not in Valid State, Port must not be a 1:1 protection
port
```

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the NE.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameters

It is not possible to change the *portClass* to UNI unless valid values for the following UNI parameters are given:

- *clientNodeID*
- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:
`IDNV – Input, Data Not Valid, Valid values for UNI parameters required`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameters not settable for ports of *portClass* = UNI

The following parameter is not changeable to value FIXED for ports with *portClass* = UNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It shall not be possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If lower-order cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is **any member** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI with existing cross-connections

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI while *portControl* is not LOCKOUT

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

It is possible to modify the *trbmd* for an ENNI port only

- if the *portControl* state is LOCKOUT or
- during the transition of the *portClass* to ENNI

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is unique within the system.

Error code:

`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`

Related TL1 messages

"RTRV-OC3: Retrieve OC3 port" (p. 3-2602)

References

GR-199-CORE (ED-*rr*)

ED-OC3 autonomous notifications

LOS for transmission port OC3 clears

If LOS is cleared and current port mode is AUTO, then portmode is updated (new value: MON). This update is reported via ED-OC3 notification with parameter *pmode*.

Change of the constituent rates status list

If *trbmd* = ADAPTIVE a change of the attribute *crstat* is notified by the related ED-OC3 notification.

Change of neighbor node and neighbor port information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-OC3 notification.

ED-OC3 notification

Output format

Syntax: **`ED-OC3`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OC3 port.
   Type: AID.
   Range: SPortAID of OC3 port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Description Bursty Interval. Measurement period for determining bursty errors. This parameter applies when sdsfmode=BURST.
   Type: numeric value set
   Range: 2 to 10
   Default: 7
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when sdsfmode=BURST.
   Type: numeric value set (as percentage) 5, 10, … 50
   Default: 30
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future. This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.

6. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier. The parameter expstrc specifying the expected path trace is only relevant for the section trace supervision when the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
Type: Hexadecimal digits
Range: 16 byte string Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

7. **`felpm`** – Far-end Line PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default)

8. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range: ENABLE: set a forced DUS/DNU, DISABLE (default): clear a forced DUS/DNU

9. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth. If trbmd is FIXED, then inputsig constrains the cross-connection rates that can be entered for the port. If trbmd is ADAPTIVE, then inputsig is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set
    Range: NX-NX-NX-. N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C
    Default: 31
    Example: An OC3 signal might be represented by 31 which implies 3 STS1's or 13 which implies 1 STS3C

11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONE, SDH
    Default: As specified via istddflton system level, see definition in RTRV-EQPT (system)

12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 or STS3 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default) or STS3.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID` –** Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET:

    - STM1: 28800
    - OC3: 154.

28. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

29. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM1: 2400
    - OC3: 155.
31. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
32. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
33. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.
34. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

35. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

36. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

38. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.

Range:

- ENABLE: enable consequent action for TIM
- DISABLE (default): disable consequent action for TIM.

39. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
40. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
41. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
42. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
43. **`sfthr`** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.
44. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

45. **`sonet`** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see "RTRV-STM4: Retrieve STM4 port" (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
Default: DEFAULT.

46. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
51. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
52. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does

not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
Default: 3 1 = all STS1.

53. **`z0outmode`** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Example output

```
ED-OC3:UNITE-NE:1-1-#-#-u1-1:123456:::INPUTSIG=31,UNEQUIPOSIG=31;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-OC3:1-1-#-#-1-1::inputsig=31,unequiposig=31:”
;
```

ED-OC12: Edit OC12 port

The command **ED-OC12** is used to provision all parameters related to optical OC12 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-OC12**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC12 port.
   Type: AID.
   Range: OC12 port AID
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.

3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

6. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
7. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
8. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is setable and retrievable as in fixed mode but does not influence the port behavior.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C, STS12C.
   Default: 121.
   Example: An OC12 signal might be represented by 31-33 which implies 3 STS1's, 3 STS3C's.
9. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SONET, SDH.
   Default: As specified via istddflton system level, see definition in RTRV-EQPT (system).
10. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

11. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
12. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
13. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3 or STS12 (default)

14. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3 or STS12.

15. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN

16. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

17. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If

ndetmd = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

18. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

19. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

20. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

21. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

22. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

23. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

24. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:

    - STM4: 115200
    - OC12: 615.

25. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

26. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

27. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 2400
    - OC12: 616.

28. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

29. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

30. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.

31. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

32. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

33. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

34. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

35. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
- ENABLE: enable consequent action for TIM
- DISABLE (default): disable consequent action for TIM.

36. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
- ENABLE: enable TIM Monitor
- DISABLE (default): disable TIM Monitor

37. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.

Range:

- BURST
- POISSON (default).

38. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

39. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.

Range:

- INSERVICE (default): Provisioning values for service are accepted
- OUTOFSERVICE: Provisioning values for service are not accepted.

40. **sfthr** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

41. **sltca** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

42. **sonet** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see “RTRV-STM4: Retrieve STM4 port” (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen (“-”) is allowed in the name additionally.
Default: DEFAULT.

43. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
44. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
45. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
46. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

47. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
48. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
49. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
Default: 121 = all STS1 (fixed).

50. **`z0outmode`** – Mode of outgoing Z0 byte insertion
    The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
    Type: Alphanumeric value set.
    Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Output format

If the ED-OC12 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OC12 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SONET to SDH has the following impact related to Fault Management:

- all SONET alarms related to that port are cleared
- the corresponding SDH alarms are raised.

Switching of tributary mode

When changing the parameter *trbmd* – tributary mode from FIXED to ADAPTIVE the parameter *crstat* may change its value, too, depending on the current input signal structure. This change is notified using the related ED-OC12 notification.

When changing the parameter *trbmd* – tributary mode from ADAPTIVE to FIXED the parameter *crstat* may change its value, too, depending on the current setting of parameter *inputsig*. This change is notified using the related ED-OC12 notification.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

The switching of *istd* should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *degthr* is set to the default value for an SDH port
- *dexcthr* is set to the default value for an SDH port
- *felpm* is mapped to *femspm* (no changes)
- *inputsig* is set to the default value

- *istd* is set to SDH
- *nelpm* is mapped to *nemspm* (no changes)
- *rstimca* is set to the default value for a SDH port
- *rstimmon* is set to the default value for a SDH port
- *sdsfmode* is set to the default value for an SDH port
- *spm* is mapped to *rspm* (no changes)
- *sonet* is mapped to *sdhpn* (no changes)
- *trbmd* is set to the default value for an SDH port
- *unequiposig* is set to its default value.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *rsmstca* is set to the default value for an SDH port
- *tcamd* is set to the default value for a SDH port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmmsfeses* is set to the default value for a SDH port
- *pmmsneses* is set to the default value for a SDH port
- *pmrsneses* is set to the default value for a SDH port.

Switching of interface standard IV:

In addition to the list of attributes above the following attributes are influenced by switching of the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- **`minNNXCRate`** is set to the default value for a SDH port
- **`maxNNXCRate`** is set to the default value for a SDH port

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value for a SDH tributary
- *brstthr* is set to the default value for a SDH tributary
- *pse* (PDI-P parameter) is set to the default value for a SDH tributary
- *sdsfmode* is set to the default value for a SDH tributary.

All other tributary parameters remain unchanged.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *degthr* is set to the default value for an SDH tributary
- *dexcthr* is set to the default value for an SDH tributary
- *ptca* is set to the default value for a SDH tributary
- *tcamd* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmhpneses* is set to the default value for an SDH tributary
- *pmhpfeses* is set to the default value for an SDH tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-OCn command all resulting parameter changes are reported by ED-STMn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-OCn command all resulting tributary parameter changes are reported by ED-VC3 notifications.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual ***portClass***

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when ***portClass*** is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when ***portClass*** is set to INNI and ***ndetmd*** is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when ***portClass*** is set to INNI and ***ndetmd*** is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*

- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-OC12:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=31-33,UNEQUIPOSIG=31-33;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-OC12:1-1-#-#-1-1::inputsig=31-33,unequiposig=31-
      33:″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`ED-OC12`** command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no test access session for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: no protection for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no path protection for *trbmd* change to ADAPTIVE

Check that *trbmd* is not changed to ADAPTIVE when tributaries of this port are used as input tributaries in a path protection group.

Error code:

`SNVS – Status, Not in Valid State, a path protection group is existing`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to *ptype* and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

Example: DENY, if a command with rr = OC3 is made on a EP155 port which is provisioned as *ptype* = STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

```
SNVS – Status, Not in Valid State, only allowed for INNI port in
lockout state
```

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

```
SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI
parameters
```

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State,` *`ndetmd`* `must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

`IDNV – Input, Data Not Valid, zero values can only be set for all neighbor information consistently`

ONNS EDGE PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 4f-BLSR group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 4f-BLSR protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following preconditions must be fulfilled:

- No cross-connections of *xconcls* = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: No modification of *trbmd* to ADAPTIVE at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from FIXED to ADAPTIVE for ports with *portClass* EDGE, while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* to FIXED at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from ADAPTIVE to FIXED for ports with *portClass* EDGE, while NN cross-connections or traditional cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* EDGEwhile *portControl* is not LOCKOUT

It is possible to modify the *trbmd* for an EDGE port only

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to EDGE

Error code:

```
SNVS – Status, Not in Valid State, only allowed for EDGE port in
lockout sate
```

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters if there still any cross-connections for the EDGE port exist which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI if the port is the **protection port** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the NE.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameters

It is not possible to change the *portClass* to UNI unless valid values for the following UNI parameters are given:

- *clientNodeID*
- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:
`IDNV – Input, Data Not Valid, Valid values for UNI parameters required`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameters not settable for ports of *portClass* = UNI

The following parameter is not changeable to value FIXED for ports with *portClass* = UNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It shall not be possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If lower-order cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is **any member** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI with existing cross-connections

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI while *portControl* is not LOCKOUT

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

It is possible to modify the *trbmd* for an ENNI port only

- if the *portControl* state is LOCKOUT or
- during the transition of the *portClass* to ENNI

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is unique within the system.

Error code:
`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

```
IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to
their DEFAULT values
```

Related TL1 messages

“RTRV-OC12: Retrieve OC12 port” (p. 3-2620)

References

GR-199-CORE (ED-*rr*)

ED-OC12 autonomous notifications

LOS for transmission port OC12 clears

If LOS is cleared and current port mode is AUTO, then portmode is updated (new value: MON). This update is reported via ED-OC12 notification with parameter pmode.

Change of the constituent rates status list

If *trbmd* = ADAPTIVE a change of the attribute *crstat* is notified by the related ED-OC12 notification.

Change of neighbor node and neighbor port information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-OC12 notification.

ED-OC12 notification

Output format

The ED-OC12 notification uses the general format of a **`REPT DBCHG`** (Report Database Change) notification.

Syntax: **`ED-OC12`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OC12 port.
   Type: AID.
   Range: SPortAID of OC12 port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
6. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
7. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
8. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is setable and retrievable as in fixed mode but does not influence the port behavior.

Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C, STS12C.
Default: 121.
Example: An OC12 signal might be represented by 31-33 which implies 3 STS1's, 3 STS3C's.

9. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
Type: Alphanumeric value set.
Range: SONET, SDH.
Default: As specified via istddflton system level, see definition in RTRV-EQPT (system).

10. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

11. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

12. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

13. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).

14. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: STS1, STS3 or STS12 (default)

15. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: STS1 (default), STS3 or STS12.

16. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
17. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
18. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

19. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.
20. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
21. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
22. **`nelpm`** – Near-End Line PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

23. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

24. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
    - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
  0000000000000000000000000000000000000000…
  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

25. **pmmsfeses** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:
    - STM4: 115200
    - OC12: 615.
26. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-4/OC-12: 1 – 768000.

Default: The default values differ for SDH and SONET and the different rates:

- STM4: 115200
- OC3: 615.

27. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

28. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM4: 2400
    - OC12: 616.

29. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

30. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

31. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.

32. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

33. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

34. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

35. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: ″″
36. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
   - ENABLE: enable consequent action for TIM
   - DISABLE (default): disable consequent action for TIM.
37. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor
38. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
   - BURST
   - POISSON (default).
39. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

40. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
41. **sfthr** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.
42. **sltca** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
43. **sonet** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see "RTRV-STM4: Retrieve STM4 port" (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
Default: DEFAULT.
44. **spm** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).
45. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the

same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

46. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
47. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the *strcwfmt* parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex
48. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

49. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
   - ADAPTIVE
   - FIXED.

   Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
50. **unequiposig** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
Default: 121 = all STS1 (fixed).
51. **z0outmode** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
   - FIXEDCC (default): for inserting the fixed value CChex
   - STSID: for inserting the STS-Id of the current frame.

Example output

```
ED-OC12:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=31-33,UNEQUIPOSIG=31-33;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-OC12:1-1-#-#-1-1::inputsig=31-33,unequiposig=31-
      33:"
;
```

ED-OC48: Edit OC48 port

The command **ED-OC48** is used to provision all parameters related to optical OC48 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-OC48**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC-48 port.
   Type: AID.
   Range: OC-48 port AID
   For changing the *portControl* parameter the AL’ qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

6. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
7. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
8. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
   Default: 481.
   Example: An OC-48 signal might be represented by 91-13-112-241 which implies 9 STS1's, 1 STS3C, 1 STS12C, 24 STS1's.
9. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SDH.
   Default: SDH
10. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

11. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
12. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
13. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12 or STS48 (default)

14. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12 or STS48.
15. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
16. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
17. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If

ndetmd = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

18. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

19. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

20. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

21. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

22. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

23. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

24. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold

This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.

Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800

- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

25. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

26. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

27. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000

Default: The default values differ for SDH and SONET and the different rates:

- SDH (all rates): 2400
- SONET OC48: 2392
- SONET OC192: 8554
- SONET OC768: 22776.

28. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

29. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

30. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

31. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

32. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
33. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
34. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
    Default: empty string: *""*
35. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.

36. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
37. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
38. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
39. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
40. **`sfthr`** – Signal Failure Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –3; –4; –5.
    Default: –3.
41. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.

42. **sonet** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see "RTRV-STM16: Retrieve STM16 port" (p. 3-2899), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

43. **spm** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

44. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

45. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)

- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

46. **strcout** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
47. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
48. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400) command.

49. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
Default: 481 = all STS1 (fixed).
50. **`z0outmode`** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Output format

If the ED-OC48 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OC48 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SONET to SDH has the following impact related to Fault Management:

- all SONET alarms related to that port are cleared
- the corresponding SDH alarms are raised.

Switching of tributary mode

When changing the parameter *trbmd* – tributary mode from FIXED to ADAPTIVE the parameter *crstat* may change its value, too, depending on the current input signal structure. This change is notified using the related ED-OC48 notification.

When changing the parameter *trbmd* – tributary mode from ADAPTIVE to FIXED the parameter *crstat* may change its value, too, depending on the current setting of parameter *inputsig*. This change is notified using the related ED-OC48 notification.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

The switching of *istd* should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *degthr* is set to the default value for an SDH port
- *dexcthr* is set to the default value for an SDH port
- *felpm* is mapped to *femspm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SDH
- *nelpm* is mapped to *nemspm* (no changes)
- *rstimca* is set to the default value for a SDH port
- *rstimmon* is set to the default value for a SDH port
- *sdsfmode* is set to the default value for an SDH port
- *spm* is mapped to *rspm* (no changes)
- *sonet* is mapped to *sdhpn* (no changes)
- *trbmd* is set to the default value for an SDH port
- *unequiposig* is set to its default value.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *rsmstca* is set to the default value for an SDH port
- *tcamd* is set to the default value for a SDH port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmmsfeses* is set to the default value for a SDH port
- *pmmsneses* is set to the default value for a SDH port
- *pmrsneses* is set to the default value for a SDH port.

Switching of interface standard IV:

In addition to the list of attributes above the following attributes are influenced by switching of the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- **`minNNXCRate`** is set to the default value for a SDH port
- **`maxNNXCRate`** is set to the default value for a SDH port

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value for a SDH tributary
- *brstthr* is set to the default value for a SDH tributary
- *pse* (PDI-P parameter) is set to the default value for a SDH tributary
- *sdsfmode* is set to the default value for a SDH tributary.

All other tributary parameters remain unchanged.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *degthr* is set to the default value for an SDH tributary
- *dexcthr* is set to the default value for an SDH tributary
- *ptca* is set to the default value for a SDH tributary
- *tcamd* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmhpneses* is set to the default value for an SDH tributary
- *pmhpfeses* is set to the default value for an SDH tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-OCn command all resulting parameter changes are reported by ED-STMn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-OCn command all resulting tributary parameter changes are reported by ED-VC3 notifications.

For efficiency reasons the ED-VC3 notifications is sent out as a single notification using a range over the aid which depends on the port rate of the associated port, i. e. the Simple_STS1NumAID field in the STS1NumAID is set to

- 1 – 3 in case of an STS1/OC3 port
- 1 – 12 in case of an STS4/OC12 port
- 1 – 48 in case of an STS16/OC48 port
- 1 – 192 in case of an STS64/OC192 port
- 1 – 768 in case of an STS256/OC768 port.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-OC48:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-OC48:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:”
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the ED-OC48 command.

Validation rule: Enough bandwidth reserved for 2F protection group

Check of the tributary rate (*inputsig*, *unequiposig*) if the port is member of a 2f protection group. Then the assigned bandwith is max. half of the port's bandwith.

Error code:

`IDNV – Input, Data Not Valid, Assigned bandwidth exceeds half port's bandwith`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no test access session for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:
`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: no protection for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when this port is in an protection group.

Error code:
`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no path protection for *trbmd* change to ADAPTIVE

Check that *trbmd* is not changed to ADAPTIVE when tributaries of this port are used as input tributaries in a path protection group.

Error code:
`SNVS – Status, Not in Valid State, a path protection group is existing`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:
`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: 'ALL' qualifier used for wrong parameters

The 'ALL' qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, 'ALL' qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:
`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:
`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:
`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*

- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, ndetmd must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

```
IDNV – Input, Data Not Valid, zero values can only be set for all
neighbor information consistently
```

ONNS EDGE PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 4f-BLSR group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 4f-BLSR protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following preconditions must be fulfilled:

- No cross-connections of *xconcls* = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: No modification of *trbmd* to ADAPTIVE at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from FIXED to ADAPTIVE for ports with *portClass* EDGE, while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* to FIXED at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from ADAPTIVE to FIXED for ports with *portClass* EDGE, while NN cross-connections or traditional cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* EDGEwhile *portControl* is not LOCKOUT

It is possible to modify the *trbmd* for an EDGE port only

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to EDGE

Error code:

```
SNVS – Status, Not in Valid State, only allowed for EDGE port in
lockout sate
```

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters if there still any cross-connections for the EDGE port exist which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI if the port is the **protection port** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the NE.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameters

It is not possible to change the *portClass* to UNI unless valid values for the following UNI parameters are given:

- *clientNodeID*
- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Valid values for UNI parameters required`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:

`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameters not settable for ports of *portClass* = UNI

The following parameter is not changeable to value FIXED for ports with *portClass* = UNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It shall not be possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If lower-order cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is **any member** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI with existing cross-connections

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI while *portControl* is not LOCKOUT

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

It is possible to modify the *trbmd* for an ENNI port only

- if the *portControl* state is LOCKOUT or
- during the transition of the *portClass* to ENNI

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is unique within the system.

Error code:
`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

```
IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to
their DEFAULT values
```

Related TL1 messages

“RTRV-OC48: Retrieve OC48 port” (p. 3-2638)

References

GR-199-CORE (ED-*rr*)

ED-OC48 autonomous notifications

LOS for transmission port OC48 clears

If LOS is cleared and current port mode is AUTO, then portmode is updated (new value: MON). This update is reported via ED-OC48 notification with parameter *pmode*.

Change of the constituent rates status list

If *trbmd* = ADAPTIVE a change of the attribute *crstat* is notified by the related ED-OC48 notification.

Change of neighbor node and neighbor port information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-OC48 notification.

ED-OC48 notification

Output format

Syntax: **`ED-OC48`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-48 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Bursty Interval. Measurement period for determining bursty errors. This parameter applies when sdsfmode=BURST.
   Type: numeric value set
   Range: 2 to 10.
   Default: 7
3. **`brstthr`** – Bursty Error Threshold
   Description Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set (as percentage) 5, 10, … 50
   Default: 30
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future. This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48 representing STS1, STS3C, STS12C, STS48C.
6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier. The parameter expstrc specifying the expected path trace is only relevant for the section trace supervision when the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)
8. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range: ENABLE: set a forced DUS/DNU, DISABLE (default): clear a forced DUS/DNU.

9. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth. If *trbmd* is FIXED, then inputsig constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then inputsig is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set
    Range: NX-NX-NX-… N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
    Default: 481
    Example: An OC-48 signal might be represented by 91-13-112-241 which implies 9 STS1's, 1 STS3C, 1 STS12C, 24 STS1's.

11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONET, SDH
    Default: As specified via istddflton system level, see definition in RTRV-EQPT (system)

12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12 or STS48 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12 or STS48.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:
- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:
- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:
- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.

    Type: numeric value set (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000

    Default: The default values differ for SDH and SONET and the different rates:

    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340

28. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.

    Type: numeric value set (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000

    Default: The default values differ for SDH and SONET and the different rates:

    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

29. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Typ:e alphanumeric value set
    Range:
    - AUTO: indicates that, when a signal is detected at the port, then alarm monitoring will start (default).
    - MON: indicates that alarm monitoring is taking place on the port.
    - NMON: means that alarm monitoring is not occurring on the port.
30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
    Type: numeric value set (superset of allowed ranges for SDH and SONET): 1 – 64000
    Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8854
    - SONET OC768:22776
31. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
32. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

33. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

34. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

35. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

36. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

38. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0). This parameter is relevant only if the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE: enable consequent action for TIM, DISABLE (default): disable consequent action for TIM

39. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0). This parameter is relevant only if the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE: enable TIM Monitor, DISABLE (default): disable TIM Monitor.

40. **sdsfmode** – Signal Degrade/Signal Fail Mode
he degraded state is detected in Burst or Poisson mode.
Type alphanumeric value set
Range: BURST, POISSON (default)

41. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when sdsfmode=POISSON.
Type: numeric value set (as exponents of 10): -5, -6 (default), -7, -8, -9

42. **servcond** – Service Condition
This parameter shall indicate its status concerning service involvement.
Type alphanumeric value set
Range: INSERVICE (default): Provisioning values for service is accepted, OUTOFSERVICE: Provisioning values for service is not accepted.

43. **sfthr** – Signal Failure Threshold
Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when sdsfmode=POISSON
Type: numeric value set (as exponent of 10): -3 (default), -4, -5

44. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string
    Range: String with maximum of 24 characters
    Default: DEFAULT0

45. **`sonet`** – SONET ASAP
    SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see RTRV-STM16, output parameter sdhpn) to be compatible with current generation applications.
    Type: Alphanumeric string A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT

46. **`spm`** – Section PM Enable
    Description Enable or disable regenerator section (section) performance monitoring.
    Type; alphanumeric value set
    Range: ENABLE, DISABLE (default)

47. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming regenerator section/section trace (strc) and the expected regenerator section/section trace (*exprstrc*). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.

Range:

- 1 for specific repeating byte (SONET)
- 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
- 16 for 16 byte specific string (ITU/ETSI mode 1)
- 64 for 64 byte string (SONET).

Default: 2

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter. If ITU/ETSI mode 2 is selected (*strcwfmt* = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits
Range: 1 byte string; 16 bytes string; 64 bytes string dependent on the *strcwfmt* parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing regenerator section/section trace (*strcout*). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1)
    - 64 for 64 byte string (SONET).

51. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED
Default: The initial value is defined as the trbmddflt parameter on system level, see definition in RTRV-EQPT (system) command.

52. **`unequiposig`** – Unequipped Output Signal List
Tributary Unequipped Output Signal Rate List. This parameter is a list of STSn rates across the port’s bandwidth. It determines the rate(s) of the “unequipped” signal that will be transmitted if there is no cross-connection. unequiposig and inputsig do not need to have the identical setting. Mixed rates for tx and rx side of

the port is allowed. If trbmd is ADAPTIVE, then unequiposig is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set. NX-NX-NX-… N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
Default: 481 = all STS1 (fixed).

53. **`z0outmode`** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3,12,48).
Type: Alphanumeric value set.
Range: FIXEDCC (default). for inserting the fixed value CChex, STSID: for inserting the STS-Id of the current frame Default FIXEDCC

Example output

```
ED-OC48:UNITE-NE:1-1-#-#-u1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-OC48:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:″
;
```

ED-OC48T: Edit OC48 transparent port

The command **ED-OC48T** is used to provision all parameters related to optical OC48 transport port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-OC48T**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Type: AID.
   Range: OC-48T port AID.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **exprstrc** – Value of Expected J0
   For ED-OC48T this parameter refers to the incoming (ingress) direction.
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.

Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
3. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET.
   Type: Alphanumeric value set.
   Range: SONET.
4. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
5. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   For ED-OC48T this parameter refers to the incoming (ingress) direction.
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000

   Default: The default values differ for SDH and SONET and the different rates:

   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.

6. **`pmrsnesesout`** – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:

   The default values differ for SDH and SONET and the different rates:

   - SDH (all rates): 2400
   - SONET OC48: 2392
7. **`rstimmon`** – RS TIM Monitor Enable
   For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
   Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
   This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
   Type: Alphanumeric value set.

   Range:

   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor
8. **`rstimmonout`** – RS TIM Monitor Enable in Outgoing direction
   Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor
9. **`servcond`** – Service Condition
   This parameter indicates its status concerning service involvement.
   Type: Alphanumeric value set.

   Range:

   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

10. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
11. **`sonet`** – SONET ASAP
    This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see "RTRV-STM16: Retrieve STM16 port" (p. 3-2899), output parameter *sdhpn*) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
12. **`spm`** – Section PM Enable
    For ED-OC48T this parameter refers to the incoming (ingress) direction.
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
13. **`spmout`** – Section PM Enable in Outgoing Direction
    Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
14. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

    For ED-OC48T this parameter refers to the incoming (ingress) direction.

15. **`strcfmtout`** – Format of outgoing J0
    The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
    Type: Numeric string.
    Range:
    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).

Output format

If the ED-OC48T command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OC48T notification for port (except for parameter *istd;* see below).

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt/strcfmtout) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc/expstrcout) and the currently received J0 (strc/strcout).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SONET to SDH has the following impact related to Fault Management:

- all SONET alarms related to that port are cleared
- the corresponding SDH alarms are raised.

Switching of interface standard – Impact on port parameters

The switching of *istd* should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard :

In detail the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *istd* is set to SDH
- *rstimmon* is set to the default value for a SDH port
- *rstimmonout* is set to the default value for a SDH port
- *spm* is mapped to *rspm* (no changes)
- *spmout* is mapped to rspmout (no changes)
- *sonet* is mapped to *sdhpn* (no changes)
- *rsmstca* is set to the default value for an SDH port
- *tcamd* is set to the default value for a SDH port
- *pmrsneses* is set to the default value for a SDH port
- *pmrsnesesout* is set to the default value for a SDH port.

All other port parameters remain unchanged.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-OCn command all resulting parameter changes are reported by ED-STMn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged

3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the impacts listed above, the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Example input/output

```
ED-OC48T:UNITE-NE:1-1-#-#-1-1:123456:::SPM=ENABLE,SPMOUT=ENABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-OC48T:1-1-#-#-1-1::spm=ENABLE,spmout=ENABLE:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ED-OC48T** command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:
`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:
`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **ED** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **ED** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Check CCB against *ptype* and *istd*

Check wether the CCB fits to ptype and istd.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

Related TL1 messages

“RTRV-OC48T: Retrieve OC48T port” (p. 3-2656)

References

GR-199-CORE (ED-*rr*)

ED-OC48T autonomous notifications

LOS for OC48 Transparent transmission port clears

If LOS is cleared and current port mode is AUTO, then *portmode* is updated (new value: MON). This update is reported via ED-OC48T notification with parameter *pmode*.

ED-OC48T notification

Output format

Syntax

`ED-OC48T`*:aid::spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-48 port AID.

Spec block parameters

1. **`exprstrc`** – Value of Expected J0
   For ED-OC48T Notification this parameter refers to the incoming (ingress) direction.
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

3. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
4. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET.
   Type: Alphanumeric value set.
   Range: SONET.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

7. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   For ED-OC48T Notification this parameter refers to the incoming (ingress) direction.
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
8. **`pmrsnesesout`** – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
9. **`rstimmon`** – RS TIM Monitor Enable
   For ENT-OC48T Notification this parameter refers to the incoming (ingress) direction.
   Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
   This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
   Type: Alphanumeric value set.
   Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor

10. **`rstimmonout`** – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor
11. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
12. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
13. **`sonet`** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see "RTRV-STM16: Retrieve STM16 port" (p. 3-2899), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
14. **`spm`** – Section PM Enable
For ED-OC48T this parameter refers to the incoming (ingress) direction.
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
15. **`spmout`** – Section PM Enable in Outgoing Direction
Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).

16. **`strcfmt`** – Format of incoming J0
    For ED-OC48T this parameter refers to the incoming (ingress) direction.
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
17. **`strcfmtout`** – Format of outgoing J0
    The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
    Type: Numeric string.
    Range:
    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).

Example input/output

```
ED-OC48T:UNITE-NE:1-1-#-#-1-1:123456:::SPM=ENABLE,SPMOUT=ENABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-OC48T:1-1-#-#-1-1::spm=ENABLE,spmout=ENABLE:”
;
```

ED-OC192: Edit OC192 port

The command **ED-OC192** is used to provision all parameters related to an optical OC192 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-OC192**:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC192 port.
   Type: AID.
   Range: SPortAID of OC192 port.
   The keyword **all** is allowed up to shelf level.
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.

3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

6. **`fec`** – FEC Enable
   Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
   Type: Alphanumeric value set.
   Range:
   DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768/STM256 basis low cost intra office 1310 nm optics.

   ENABLE / DISABLE for:
   - OC192 / STM64 OLS800G compatible 80 colour optics
   - OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
   - OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
   - OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

   Default:
   DISABLE for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768/STM256 basis low cost intra office 1310 nm optics

   ENABLE for:
   - OC192 / STM64 OLS800G compatible 80 colour optics
   - OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
   - OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
   - OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)
7. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

8. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.

   Range:

   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
   Default: 1921.

10. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: SDH

11. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

12. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass*

values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

13. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

14. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: STS1, STS3, STS12, STS48 or STS192 (default)

15. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: STS1 (default), STS3, STS12, STS48 or STS192.

16. **`msrei`** – MS-REI mode
Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
Type: Alphanumeric value set.
Range:
    - SINGLE
    - EXTENDED (default).
17. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN
18. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

19. **neighborNodeID** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

20. **neighborPortID** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

21. **neighborSigPCID** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

22. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

24. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
   - IPV4 (default)
   - IPV6
   - NSAP

25. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

26. **och** – optical channel enable
Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don’t have any impact on transmission.
Type: Alphanumeric value set.

Range:

- for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
- for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel

- for ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
- for client ports on packs which support OTN: DISABLE; disable, optical channel not applicable.

Default:

- for OP10 and Op40 packs, which cannot support OCH: DISABLE
- for OP10 packs, which cannot support OCH: ENABLE
- for OP10 and OP40 packs, which only support OCH: ENABLE
- for client ports on packs which support OTN: DISABLE.

27. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

28. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

29. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

    Default:
    - AUTO for optical SDH/SONET ports
    - NMON for client ports.
30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.

31. **`polynomial`** – Optical Channel Scrambling Polynomial
    This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
    Type: numeric value set.
    Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
    Default: 2.
32. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
33. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
34. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.
35. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

36. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295
37. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0
38. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: *""*
39. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.

40. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
41. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
42. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
43. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
44. **`sfthr`** – Signal Failure Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –3; –4; –5.
    Default: –3.
45. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.

46. **`sonet`** – SONET ASAP
    This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see “RTRV-STM64: Retrieve STM64 port” (p. 3-2926), output parameter *sdhpn*) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
47. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
48. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
49. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)

- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

50. **strcout** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
51. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
52. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.

Range:

- ADAPTIVE
- FIXED.

Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

53. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12, 48 or 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
Default: 1921 = all STS1 (fixed).

Output format

If the ED-OC192 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OC192 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SONET to SDH has the following impact related to Fault Management:

- all SONET alarms related to that port are cleared
- the corresponding SDH alarms are raised.

Switching of tributary mode

When changing the parameter *trbmd* – tributary mode from FIXED to ADAPTIVE the parameter *crstat* may change its value, too, depending on the current input signal structure. This change is notified using the related ED-OC192 notification.

When changing the parameter *trbmd* – tributary mode from ADAPTIVE to FIXED the parameter *crstat* may change its value, too, depending on the current setting of parameter *inputsig*. This change is notified using the related ED-OC192 notification.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

The switching of *istd* should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *degthr* is set to the default value for an SDH port
- *dexcthr* is set to the default value for an SDH port
- *felpm* is mapped to *femspm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SDH
- *nelpm* is mapped to *nemspm* (no changes)
- *rstimca* is set to the default value for a SDH port
- *rstimmon* is set to the default value for a SDH port
- *sdsfmode* is set to the default value for an SDH port
- *spm* is mapped to *rspm* (no changes)
- *sonet* is mapped to *sdhpn* (no changes)
- *trbmd* is set to the default value for an SDH port
- *unequiposig* is set to its default value.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *rsmstca* is set to the default value for an SDH port
- *tcamd* is set to the default value for a SDH port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmmsfeses* is set to the default value for a SDH port
- *pmmsneses* is set to the default value for a SDH port
- *pmrsneses* is set to the default value for a SDH port.

Switching of interface standard IV:

In addition to the list of attributes above the following attributes are influenced by switching of the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- **`minNNXCRate`** is set to the default value for a SDH port
- **`maxNNXCRate`** is set to the default value for a SDH port

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SONET to SDH (see also the related RTRV-STM*n* command for SDH parameter description):

- *brstintvl* is set to the default value for a SDH tributary
- *brstthr* is set to the default value for a SDH tributary
- *pse* (PDI-P parameter) is set to the default value for a SDH tributary
- *sdsfmode* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *degthr* is set to the default value for an SDH tributary
- *dexcthr* is set to the default value for an SDH tributary
- *ptca* is set to the default value for a SDH tributary
- *tcamd* is set to the default value for a SDH tributary.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH (see also the related RTRV-STMn command for SDH parameter description):

- *pmhpneses* is set to the default value for an SDH tributary
- *pmhpfeses* is set to the default value for an SDH tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-OCn command all resulting parameter changes are reported by ED-STMn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-OCn command all resulting tributary parameter changes are reported by ED-VC3 notifications.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while ***portClass*** is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual ***portClass***

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when ***portClass*** is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-OC192:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-OC192:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-OC192 command.

Validation rule: Enough bandwidth reserved for 2F protection group

Check of the tributary rate (*inputsig*, *unequiposig*) if the port is member of a 2f protection group. Then the assigned bandwith is max. half of the port’s bandwith.

Error code:
`IDNV – Input, Data Not Valid, Assigned bandwidth exceeds half port’s bandwith`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections for fixed tributary mode

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross connections-connections if the port operates in the fixed tributary mode.

Error code:

`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no test access session for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: no protection for *istd*, *trbmd* change

Check that *istd* or *trbmd* is not set to FIXED when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no path protection for *trbmd* change to ADAPTIVE

Check that *trbmd* is not changed to ADAPTIVE when tributaries of this port are used as input tributaries in a path protection group.

Error code:

`SNVS – Status, Not in Valid State, a path protection group is existing`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode`.

Validation rule: Change of FEC parameters only if supported by the current circuit pack

When changing one of the FEC-related parameters *fec*, *feccor* and *fectype*, it is checked whether the currently provisioned circuit pack supports this. If it is not supported the following error code is used:

Error code: `IDNV – Input, Data Not Valid, Parameter not supported`.

Validation rule: Enabling of Optical Channel only if supported by the current circuit pack

When enabling the optical channel via parameter *och*, it is checked whether the currently provisioned circuit pack supports this. If it is not supported the following error code is used:

Error code: `IDNV – Input, Data Not Valid, Parameter not supported`.

Validation rule: IDNV - Input, Data Not Valid, Parameter not supported

The command is denied if FEC is enabled (*fec* = ENABLE) while the optical channel is disabled or if the optical channel is disabled (*och* = DISABLE) while FEC is enabled.

It is allowed to enable or disable FEC and the optical channel together in one command.

Error code:

`SNVS – Status, Not in Valid State, Optical Channel must be enabled.`

Validation rule: 'ALL' qualifier used for wrong parameters

The 'ALL' qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:

`IDNC – Input, Data Not Consistent, 'ALL' qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with *rr* = OC3 is made on a EP155 port which is provisioned as *ptype* = STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:
`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:
`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:
`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*

- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, *ndetmd* must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

```
IDNV – Input, Data Not Valid, zero values can only be set for all
neighbor information consistently
```

ONNS EDGE PORT related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 4f-BLSR group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 4f-BLSR protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following preconditions must be fulfilled:

- No cross-connections of *xconcls* = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: No modification of *trbmd* to ADAPTIVE at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from FIXED to ADAPTIVE for ports with *portClass* EDGE, while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* to FIXED at ports of *portClass* EDGE with existing ONNS cross-connections

The system shall not allow to set the tributary mode (parameter *trbmd*) from ADAPTIVE to FIXED for ports with *portClass* EDGE, while NN cross-connections or traditional cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* EDGEwhile *portControl* is not LOCKOUT

It is possible to modify the *trbmd* for an EDGE port only

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to EDGE

Error code:

```
SNVS – Status, Not in Valid State, only allowed for EDGE port in
lockout sate
```

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters if there still any cross-connections for the EDGE port exist which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI if the port is the **protection port** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI if UNI-N node is not enabled for the NE.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameters

It is not possible to change the *portClass* to UNI unless valid values for the following UNI parameters are given:

- *clientNodeID*
- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Valid values for UNI parameters required`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:

`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameters not settable for ports of *portClass* = UNI

The following parameter is not changeable to value FIXED for ports with *portClass* = UNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It shall not be possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist

Check if lower-order cross-connections, substructure, or conversion cross-connections exist which involve the port to be changed. If lower-order cross-connections, substructure, or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is the **protection port** of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI if the port is **any member** of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI with existing cross-connections

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No modification of *trbmd* at ports of *portClass* ENNI while *portControl* is not LOCKOUT

The tributary mode (parameter *trbmd*) is not changeable for ports with *portClass* ENNI while cross-connections exist at that port.

It is possible to modify the *trbmd* for an ENNI port only

- if the *portControl* state is LOCKOUT or
- during the transition of the *portClass* to ENNI

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is unique within the system.

Error code:
`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`

Related TL1 messages

"RTRV-OC192: Retrieve OC192 port" (p. 3-2664)

References

GR-199-CORE (ED-*rr*)

ED-OC192 autonomous notifications

LOS for transmission port OC192 clears

If LOS is cleared and current port mode is AUTO, then portmode is updated (new value: MON). This update is reported via ED-OC192 notification with parameter *pmode*.

Change of the constituent rates status list

If *trbmd* = ADAPTIVE a change of the attribute *crstat* is notified by the related ED-OC192 notification.

Change of neighbor node and neighbor port information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-OC192 notification.

ED-OC192 notification

Output format

Syntax: **`ED-OC192`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OC192 port.
   Type: AID.
   Range: SPortAID of OC192 port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Bursty Interval. Measurement period for determining bursty errors. This parameter applies when sdsfmode=BURST.
   Type: numeric value set
   Range: 2 to 10
   Default: 7
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when sdsfmode=BURST.
   Type: numeric value set (as percentage) 5, 10, … 50
   Default: 30
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the tributary input rate signal list.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C, STS192C.
6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier. The parameter expstrc specifying the expected path trace is only relevant for the section trace supervision when the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
   Type Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **`fec`** – FEC Enable
   Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
   Type: Alphanumeric value set.
   Range:
   DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768/STM256 basis low cost intra office 1310nm optics.

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

Default:

DISABLE for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768/STM256 basis low cost intra office 1310 nm optics

ENABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

8. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)

9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range: ENABLE (default): set a forced DUS/DNU), DISABLE: clear a forced DUS/DNU.

10. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

Range:

- YES
- NO (default).

11. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of STSn rates across the port's bandwidth. If *trbmd* is FIXED, then inputsig constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then inputsig is settable and retrievable as in fixed mode but does not influence the port behavior.
Type numeric value set
Range: NX-NX-NX-… N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C,STS48C or STS192C
Default: 1921

12. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET.
Type: Alphanumeric value set.
Range: SONET, SDH
Default: As specified via istddflton system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400)

13. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

14. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

15. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
16. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
17. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12, STS48 or STS192 (default)

18. **minNNXCRate** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: STS1 (default), STS3, STS12, STS48 or STS192.

19. **msrei** – MS-REI mode
Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
Type: Alphanumeric value set.
Range:
    - SINGLE
    - EXTENDED (default).

20. **ndetmd** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **neighborASTNNodeID** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID` –** Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:
- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:
- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:
- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`och`** – Optical Channel enable
Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don’t have any impact on transmission.
Type: Alphanumeric value set.

Range:

- for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
- for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE disable optical channel
- for ports on OP10 and OP40 packs, which only support OCH: ENABLE enable optical channel
- for client ports on packs which support OTN: DISABLE: disable, optical channel not applicable.

Default:

- for OP10 and OP40 packs, which cannot support OCH: DISABLE
- for OP10 packs, which can support OCH: ENABLE
- for OP10 and OP40 packs, which only support OCH: ENABLE
- for client ports on packs which support OTN: DISABLE.

30. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

31. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

32. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

Default:

- AUTO for optical SDH/SONET ports
- NMON for client ports.

33. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
Type: numeric value set (superset of allowed ranges for SDH and SONET): 1 – 64000 ·
Default: The default values differ for SDH and SONET and the different rates:

- SDH (all rates): 2400
- SONET OC48: 2392
- SONET OC192: 8854
- SONET OC768:22776

34. **`polynomial`** – Optical Channel Scrambling Polynomial
The polynomial used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE)
Type: numeric value set
Range: 1x7 + x + 1, 2x16 + x12 + x3 + x + 1
Default: 2

35. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

36. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

37. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

38. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

39. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
40. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
41. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: *""*
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0). This parameter is relevant only if the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
    Type alphanumeric value set
    Range: ENABLE: enable consequent action for TIM, DISABLE (default): disable consequent action for TIM.

43. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
45. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`sfthr`** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.
48. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

49. **`sonet`** – SONET ASAP
    This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see "RTRV-STM64: Retrieve STM64 port" (p. 3-2926), output parameter *sdhpn*) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
50. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
51. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
52. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
    Type: Numeric string.
    Default: 2
53. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string dependent on the strcwfmt parameter..

Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

54. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Default: 2

55. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
   - ADAPTIVE
   - FIXED.

   Default: FIXED. The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

56. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12, 48 or 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
Default: 1921 = all STS1 (fixed)

Example output

```
ED-OC192:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
```

```
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-OC192:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:"
;
```

ED-OTU2: Edit OTU2 port

The command **ED-OTU2** is used to edit parameters related to the OTU2 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: Yes

Input format

Command syntax: **ED-OTU2**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   Access Identifier of an OTU2 port.
   Type: AID
   Range: OTNPortAID of OTU2 port.
   The keyword ″otall″ is not allowed.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string.
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831

Spec block parameters

1. **expotutti** – Expected OTU TTI
   Expected OTU trail trace identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the expected OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte hexadecimal string
   Default: 15 bytes each with value 00.
2. **fectype** – FEC Type
   Forward error correction type.
   Type: Alphanumeric value set.
   Range: RS_FEC, V_FEC (default).

3. **`ochpm`** – OCH PM Enable
   Enable or disable optical channel performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
4. **`opwl`** – Optical Wavelength
   Optical Wavelength in nanometers.
   Type: numeric value set
   Range:
   - 35 wavelengths in C-Band (1554.537 … 1568.362), i.e:
     1554.537, 1554.940, 1555.343, 1555.747, 1556.151, 1556.555, 1556.959, 1557.363, 1557.768, 1558.173, 1558.578, 1558.983, 1559.389, 1559.794, 1560.200, 1560.606, 1561.013, 1561.419, 1561.828, 1562.233, 1562.640, 1563.047, 1563.455, 1563.863, 1564.271, 1564.679, 1565.087, 1565.496, 1565.905, 1566.314, 1566.723, 1567.133, 1567.542, 1567.952, 1568.362
   - 93 wavelengths in L-Band (1568.773 … 1607.466), i.e:
     1568.773, 1569.183, 1569.594, 1570.005, 1570.416, 1570.828, 1571.239, 1571.651, 1572.063, 1572.476, 1572.888, 1573.301, 1573.714, 1574.127, 1574.540, 1574.954, 1575.368, 1575.782, 1576.196, 1576.610, 1577.025, 1577.440, 1577.855, 1578.270, 1578.686, 1579.102, 1579.518, 1579.934, 1580.350, 1580.767, 1581.184, 1581.601, 1582.018, 1582.436, 1582.854, 1583.271, 1583.690, 1584.108, 1584.527, 1584.946, 1585.365, 1585.784, 1586.203, 1586.623, 1587.043, 1587.463, 1587.884, 1588.304, 1588.725, 1589.146, 1589.568, 1589.989, 1590.411, 1590.833, 1591.255, 1591.678, 1592.100, 1592.523, 1592.946, 1593.369, 1593.793, 1594.217, 1594.641, 1595.065, 1595.489, 1595.914, 1596.339, 1596.764, 1597.189, 1597.615, 1598.041, 1598.467, 1598.893, 1599.320, 1599.746, 1600.173, 1600.600, 1601.028, 1601.455, 1601.883, 1602.311, 1602.740, 1603.168, 1603.597, 1604.026, 1604.455, 1604.885, 1605.314, 1605.744, 1606.174, 1606.605, 1607.035, 1607.466
5. **`otutimca`** – OTU TIM Consequent Action Enable
   OTU Trail Trace Identifier Mismatch consequent action enable. The parameter *otutimca* shall enable or disable the TIM consequent action. In addition, to the TIM consequent action also the TIM detection is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE: enable consequent action for TIM (default), DISABLE: disable consequent action for TIM.
6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default):

indicates that, when a signal is detected at the port, then alarm monitoring will start.

- MON:
  indicates that alarm monitoring is taking place on the port.
- NMON:
  means that alarm monitoring is not occurring on the port.

7. **`txenable`** – Optical Output Enable
   Optical output enabling.
   Type: Alphanumeric value set.
   Range:
   - DISABLE (default): optical output disabled
   - ENABLE: optical output enabled.
8. **`votutti`** – Value of outgoing OTU TTI
   Value of the transmitted OTU Trail Trace Identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the transmitted OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte SAPI (Source Access Point Identifier).
   Default: 15 bytes each with value 00.

Output format

If the ED-OTU2 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-OTU2 notification for port.

Disabling of PM Monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. Stop PM counting
2. All current bins related to the PM monitor is deleted
3. All history bins related to the PM monitor is deleted

PM monitors can be disabled explicitly by means of the 'pm enable' attribute or are implicitly disabled as part of a port/tributary deletion.

Example input/output

```
ED-OTU2:UNITE-NE:1-1-#-#-1-1:123456:::FECTYPE=RS_FEC;

   UNITE-NE 06-06-30 08:00:00
M  123456 COMPLD
;

   UNITE-NE 06-06-30 08:00:01
A  001 REPT DBCHG
   "TIME=08-00-00,DATE=06-06-30:ED-STM16:1-1-#-#-1-1::fectype=RS_FEC:"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-OTU2 command.

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Validation rule: Output can only be enabled when wavelength is provisioned

Check that the optical wavelength is provisioned, i.e. *opwl* is set away from 0, when the optical output is enabled (*txenable* = ENABLE).

Error code:

`SNVS – Status, Not in Valid State, optical wavelength is not provisioned`

Validation rule: Optical wavlength must be in the optical window

Check that the optical wavelength (opwl) specified in the command is in the optical window (opwind) supported by the pack.

Error code:

`IDNV – Input, Data Not Valid, optical wavelength not in optical window`

Related TL1 messages

- "RTRV-OTU2: Retrieve OTU2 port" (p. 3-2685)

ED-OTU2 autonomous notification

LOS for transmission port OTU2 clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-OTU2 notification with parameter pmode.

ED-OTU2 notification

Output format

Command syntax: **ED-OTU2**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of an OTU2 port.
   Type: AID
   Range: OTNPortAID of OTU2 port.
   The keyword ″otall″ is not allowed.

Spec block parameters

1. **expotutti** – Expected OTU TTI
   Expected OTU trail trace identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the expected OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte hexadecimal string
   Default: 15 bytes each with value 00.
2. **fectype** – FEC Type
   Forward error correction type.
   Type: Alphanumeric value set.
   Range: RS_FEC, V_FEC (default).
3. **ochpm** – OCH PM Enable
   Enable or disable optical channel performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
4. **opwl** – Optical Wavelength
   Optical Wavelength in nanometers.
   Type: numeric value set
   Range:
   - 35 wavelengths in C-Band (1554.537 … 1568.362), i.e:
     1554.537, 1554.940, 1555.343, 1555.747, 1556.151, 1556.555, 1556.959, 1557.363, 1557.768, 1558.173, 1558.578, 1558.983, 1559.389, 1559.794, 1560.200, 1560.606, 1561.013, 1561.419, 1561.828, 1562.233, 1562.640, 1563.047, 1563.455, 1563.863, 1564.271, 1564.679, 1565.087, 1565.496, 1565.905, 1566.314, 1566.723, 1567.133, 1567.542, 1567.952, 1568.362
   - 93 wavelengths in L-Band (1568.773 … 1607.466), i.e:

1568.773, 1569.183, 1569.594, 1570.005, 1570.416, 1570.828, 1571.239, 1571.651, 1572.063, 1572.476, 1572.888, 1573.301, 1573.714, 1574.127, 1574.540, 1574.954, 1575.368, 1575.782, 1576.196, 1576.610, 1577.025, 1577.440, 1577.855, 1578.270, 1578.686, 1579.102, 1579.518, 1579.934, 1580.350, 1580.767, 1581.184, 1581.601, 1582.018, 1582.436, 1582.854, 1583.271, 1583.690, 1584.108, 1584.527, 1584.946, 1585.365, 1585.784, 1586.203, 1586.623, 1587.043, 1587.463, 1587.884, 1588.304, 1588.725, 1589.146, 1589.568, 1589.989, 1590.411, 1590.833, 1591.255, 1591.678, 1592.100, 1592.523, 1592.946, 1593.369, 1593.793, 1594.217, 1594.641, 1595.065, 1595.489, 1595.914, 1596.339, 1596.764, 1597.189, 1597.615, 1598.041, 1598.467, 1598.893, 1599.320, 1599.746, 1600.173, 1600.600, 1601.028, 1601.455, 1601.883, 1602.311, 1602.740, 1603.168, 1603.597, 1604.026, 1604.455, 1604.885, 1605.314, 1605.744, 1606.174, 1606.605, 1607.035, 1607.466

5. **`otutimca`** – OTU TIM Consequent Action Enable
   OTU Trail Trace Identifier Mismatch consequent action enable. The parameter *otutimca* shall enable or disable the TIM consequent action. In addition, to the TIM consequent action also the TIM detection is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE: enable consequent action for TIM (default), DISABLE: disable consequent action for TIM.
6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default):
     indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON:
     indicates that alarm monitoring is taking place on the port.
   - NMON:
     means that alarm monitoring is not occurring on the port.
7. **`txenable`** – Optical Output Enable
   Optical output enabling.
   Type: Alphanumeric value set.

Range:

- DISABLE (default): optical output disabled
- ENABLE: optical output enabled.

8. **`votutti`** – Value of outgoing OTU TTI
   Value of the transmitted OTU Trail Trace Identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the transmitted OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte SAPI (Source Access Point Identifier).
   Default: 15 bytes each with value 00.

Example input/output

```
ED-OTU2:UNITE-NE:1-1-#-#-1-1:123456:::FECTYPE=RS_FEC;

   UNITE-NE 06-06-30 08:00:00
M  123456 COMPLD
;

   UNITE-NE 06-06-30 08:00:01
A  001 REPT DBCHG
   "TIME=08-00-00,DATE=06-06-30:ED-STM16:1-1-#-#-1-1::fectype=RS_FEC:"
;
```

ED-PATH: Add or modify alias of ONNS connection

Overview

The **ED-PATH** command is used to modify the user defined alias of an ONNS connection or to add a new alias to an ONNS connection.

ED-PATH can be used to edit or replace the user defined connection alias itself. In this case the spec parameter *connAlias* identifies the old string and the spec parameter *newconnAlias* specifies the new string.

connID	*connAlias*	*newConnAlias*	meaning
present	present	present	*connAlias* replaced by *newConnAlias*, *connID* used for identification
present	not present	present	*connAlias* set to *newConnAlias*, *connID* used for identification
not present	present	present	*connAlias* replaced by *newConnAlias*, *connAlias* used for identification
not present	not present	present	DENY, inconsistent input
(not) present	(not) present	not present	DENY, inconsistent input

The **ED-PATH** command is applicable for connections with *connControl* = ONNS (e.g. ONNS connections initiated by Management System) and for connections with *connControl* = EXT (e.g. ONNS connections initiated by UNI).

Original routing constraints of ONNS connection get lost in case of manual reroute with explicit node constraint

In case an ONNS connection, that originally has been setup with routing constraints like “exclude SRLG” or “max delay”, is manually rerouted with routing constraint “explicit node”, the original routing constraints will get lost.

This will be the case even if the rerouted connection still fulfills the original constraints.

Customer workaround: After manual reroute with constraints being lost, another manual reroute can be applied specifying the original constraints again, if these are still known.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No

Input format

Command syntax:
ED-PATH:*srcTid:[srcAid]:ctag::[srvID]:[connID]:[ConnAlias=connAlias,]NewConnAlias=newConnAlias;*

Input parameters

Command parameters

1. **srcTid**
Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique Tid used by Host NE. See also the definition of tid in "RTRV-HDR: Retrieve header" (p. 3-2417).
Type: identifier, in addition the special character underscore "_" and the password characters are allowed
Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. **srcAid**
Source node ONNS application access identifier *srcAid* of the ONNS controller OS command interface port.
Type: AID
Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn

3. **ctag**
This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages. See also the definition of ctag in "RTRV-HDR: Retrieve header" (p. 3-2417).
Type alphanumeric string
Range: The ctag parameter is a character strings of up to 6 ASCII characters.The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.

4. **`srvID`**
   The service identifier (srvID) is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   Type Integer
   Range: 1.. 20,000,000
   Only one *srvID* accepted. The value ALL is not supported.
5. **`connID`** – Connection ID for an existing ONNS optical path connection service
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and Ingress Tributary# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.
   Range: *ALL* – special value to retrieve all connection records

   Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x. Example: 29.13.196.203&1-1-13-15-1&271584

   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with

   - nodeID: four dot separated numeric strings: each string 0..255;
   - Tributary AID: five dash separated numeric strings representing:

     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

   As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

   Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - default Tributary AID: five dash separated numeric strings with:

     Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##.
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Spec block parameters

1. **`connAlias`** – connection alias
   connAlias specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83.
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.
2. **`newConnAlias`** – New connection alias
   This parameter is the replacement of the existing *connAlias* or the new *connAlias* (in case no one was defined so far). There is no possibility to remove the connAlias via this command.
   Type: Character string up to length 83
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes.

Output format

If the **`ED-PATH`** request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ”ConnID=connID,ConnStatus=connStatus,pdir,,bandwidth,
   ConnAlias=newConnAlias,
   ,SrvID=srvID,,,,,specBlock,,
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID[,
   brDestNID=brdestNID,brdestPortID]:
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1[:
   PathID=pathID2,State=pst2&sst2&tst2,
   NodeList=pNodeList2]”
;
```

The returned value for *connAlias* is the replaced string.

Output parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   This parameter is set to the date when the output message is generated:
   YY-MM-DD where YY=00-99, MM=01-12, DD=01-31.
3. **`time`**
   This parameter is set to the time when the output message is generated:
   HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.
4. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages. The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
5. **`completionCode`**
   Standard TL1 completion code result.
   Type: Value set.

   Range:

   - COMPLD: for success (default)
   - PRTL: for partial success / failure, see section ERROR RESPONSES for details
   - DENY: in case of errors, see section ERROR RESPONSES for details
6. **`connID`** – Connection ID
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and IngressPort# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.
   Range: *ALL* – special value to retrieve all connection records

   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing with:

Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 .. 192).

- Index: Numeric string 0-4294967295

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x. Example: 29.13.196.203&1-1-13-15-1&271584.

Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.
The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-5-13-7-ALL.
The port AID 1-5-13-7 must denote a VCG.

Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- defaultTributary AID: five dash separated numeric strings with Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#)
- Index: Numeric string 0-4294967295.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

7. **`connStatus`** – Connection Status
   ConnStatus = pst&sst(&tst) Connection status is the overall status of the connection connID which may be different than the status of an individual *pathID* in that connection.

connStatus is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter. The two values must be separated by an ampersand (“&”). Individual status values are 7 characters or less.

Type: up to three &-separated value sets.

Range *pst* values:

- CON: Connected
- DLT: Deleted
- NOCON: Not Connected
- MAINT: Maintenence Mode
- UNKNOWN: Conn state unknown.

Range *sst* & *tst* values:

- OK: Conn good
- FAILED: Conn failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- PREMPT: Path pre-empted
- SETUPFL: Connection SetUp Failed
- FL-NVM: Not saved in NVM
- WTORV: Waiting to revert path
- EPRTFL: EDGE Port Failed
- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values:

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection)
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection)

Range *tst* values:

- DEGRADED: Non-revertive restoration was not possible if the worker or protection path of a permanent 1+1 protected connection fails.

8. **`pdir`** – Path Direction Topology
   Uni-direction A-Z (1WAY) or Bi-direction – (2WAY).
   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default), 2WAY
9. **`bandwidth`** – Requested bandwidth/signalRate
   Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
   Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
10. **`connAlias`** – connection Alias
    This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
    Type: Character string up to length 83.
    Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes.
    This are an optional output parameter in the **`ED-PATH`** command reply.
11. **`srvID`** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.
12. **`srcNID`** – Source nodeID
    ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
    Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
13. **`srcPortID`** – Source PortID
    ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
    Value: dash-separated list of 5 numeric strings

Example: 1-1-3-1-4
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups

14. **`destNID`** – Destination nodeID
ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

15. **`destPortID`** – Destination PortID
ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups

16. **`brDestNID`** – Bridged Destination NodeID
brDestNID is an optional output parameter in the **`ED-PATH`** command reply. It is used for the second path of a YCONN connection.
ONNS bridged destination nodeID. Together with the following brDestPortID field defines the second associated Egress connPoint that connects the destination node to the edge/gateway switch or client router in case of protectionClass YCONN.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

17. **`brDestPortID`** – Bridged Destination PortID
ONNS bridged destination node egress *portID* in case *protectionClass* = YCONN. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the brDestNID egress connPoint. 4.1; 5; 6
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
brDestPortID is an optional output parameter in the **`ED-PATH`** command reply. It is sed for the second path of a YCONN connection.

18. **`pathID1`** – Path ID 1
The PathID of the 1st path that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255) (Index: Numeric string).
The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111

19. **`pathState1`** – Path State 1
Path State1 = *pst*& *sst*[*&tst*]. pathState1 represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.
PathState1 is composed out of *pst* (PrimaryState), *sst* (SecondaryState) and *tst* (TertiaryState, optional) Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets.
Range *pst* values:
    - CON: Connected
    - NOCON: Not Connected
    - DLT: Deleted
    - MAINT: Maintenance mode.

    Range *sst* & *tst* values
    - OK: Conn good
    - FAILED-PATH: failed
    - SETUP: Conn SetUp occurring
    - TRDOWN: Conn teardown occurring
    - NO-IN: Conn failed, signal in bad
    - RSTOR: path restore occurring
    - RSTRD: Conn restored
    - RSTRFL: Conn restore failed
    - DLT-FL: Delete operation Failed
    - UNKNOWN: Conn state unknown
    - WTORSTR: Waiting to restore path
    - SETUPFL: Connection SetUp Failed
    - RVRTFL: Conn revert failed
    - PREMPTD: Preempted
    - WTORV: Waiting to revert path
    - BRDGD: Path bridged
    - EPRTFL: EDGE Port Failed
    - BAD-IN: path failed, Bad signal in
    - FLT: Fault.

    Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

Range *pst* values:

- RSVD: Reserved, for paths associated to protecting connections within an M:N protection group, while not protecting a working traffic.

20. **`pNodeList1`** – Path Node List 1
The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
This is an optional output parameter in the **`ED-PATH`** command reply.

21. **`pathID2`** – Path ID 2
The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255), (Index: Numeric string 0-65535).
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111
pathID2 is an optional output parameter in the **`ED-PATH`** command reply. It is used for the second path of a N1PLUS1 connection.

22. **`pathState2`** – Path State 2
pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values

- OK: Conn good
- FAILED-PATH: failed
- SETUP: Conn SetUp occurring

- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.
This is an optional output parameter in the **`ED-PATH`** command reply.

23. **`pNodeList2`** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
This are an optional output parameter in the **`ED-PATH`** command reply.

Spec block parameter

1. **`connMode`** – Connection Mode
   This parameter specifies whether revertive or non-revertive mode is used as operation mode of the auto-reroute or 1+1 protected connection.
   This parameter can be used with the **`ENT-PATH-AUTO`** and **`ENT-PATH-N1PLUS1`** command. This parameter can be used with the **`ENT-PATH-P1PLUS1NR`** command to specify, whether revertive or non-revertive mode is used as operation mode of the 1+1 protection of the permanent 1+1 protected connection.
   Type: Alphanumeric value set.
   Range: NONREVERTIVE (default), REVERTIVE
2. **`disjointness`**
   The actual level of disjointness of the calculated and established connection. The parameter is not filled when *protectionClass* is UNPROT.
   Type: Alphanumeric value set.
   Range:
   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.
3. **`rstrPriority`** – Restoration Priority
   This parameter describes the restoration priority of the connection.
   Type: Integer
   Range: 0 … 7 (lower values represent higher priority).
   Default: 0.

Example input/output

The following example shows the successful execution of the command with old and new *conn Alias*:

```
ED-PATH:DENVER:snn:1044::::ConnAlias=AE-CHG-NY,NewConnAlias=AE-BOS-NY;

   DENVER 2003-03-03 10:36:44
M  1044 COMPLD
   ″ConnID=29.13.85.0&1-1-4-1-4&159,ConnStatus=CON&OK,2WAY,,STS3,
   ConnAlias=AE-BOS-NY,
   ,SrvID=2,,,,,ConnMode=NONREVERTIVE&RstrPriority=0&Disjointness=COMPLETE,,
   SrcNID=29.13.85.0,1-1-4-1-4,DestNID=29.13.85.32,1-1-21-2-1:
   PathID=29.13.85.0-346,State=CON&OK,
   NodeList=29.13.85.0&29.13.83.176&29.13.83.192&29.13.85.32″
;
```

Error responses

ED-PATH Request Denied - Parameter InValid

The following list identifies conditions that will cause similar error responses:

Error code: `E035 – ICAE – Input, ConnAlias Already Exists`

Error code: `E037 – INCS – Input, Neither ConnID nor ConnAlias Specified`

The following error response will be raised in case old connection alias and connection ID specified in the command do not match:

Error code: `E038 – IICS – Input, Inconsistent connAlias Specified`

ConnectionID does not exist

If the Source Node ONNS receives a ED-PATH command when the specified connection path does not exist, then the following error response is returned:

Error code: `E220 – ICNE Error – Input, ConnectionID Non-Existent`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "ED-NNCNFG: Modify ONNS system configuration" (p. 3-450)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT-NNCHG notification" (p. 3-2137)
- "REPT-NNEVT notification" (p. 3-2144)
- "ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
- "ED-PATH-ROLL: Modify ONNS connection route" (p. 3-781)

ED-PATH-PROT: Modify ONNS connection parameters or protection type

Overview

The **ED-PATH-PROT** command is used to modify connection parameters or change the protection type, e.g. from unprotected to auto-reroute or from non-revertive auto-reroute to revertive auto-reroute.

ED-PATH-PROT is used to modify an existing connection

The **ED-PATH-PROT** command modifies the existing connection service based on its *connID* by changing an existing connection (or path) configuration or the connection constraints. The *reqType* parameter specifies what type of path operation(s) occurs on the connection.

The **ED-PATH-PROT** command is not applicable to ONNS connections of protectionClass MBYNWRK or MBYNPRT.

User Defined Connection Alias used as an Alternate Key

Instead of the *connID*, the user may optionally use the *connAlias* as an alternate key. The **ED-PATH-PROT** command could have either or both specified. If both are specified by the user, then *ConnID* id used as the key.

Either the *connID* or the *connAlias* must be present in a command. The *connAlias* is returned as an output parameter. If it is not defined, a NULL is returned.

ED-PATH-PROT to change a connection parameter (reqType=setParam)

The **ED-PATH-PROT** command is used to change a parameter of an ONNS connection.

With *reqType* = setParam the command can be used to change the *rstrPriority* of the connection identified by the *connID* field. The new *rstrPriority* is specified as a specBlock parameter. Changing the *rstrPriority* is allowed for connections of type *connControl* = ONNS and EXT.

With *reqTyp*e=setParam the command can be used to change the *wtr* (wait to restore timer) of the connection identified by the *connID* field. The new *wtr* is specified as a specBlock parameter for a 1+1 connection. Changing the *wtr* is allowed for connections of type *connControl* = ONNS only. Parameter applies to 1+1 protected and permanent 1+1 protected connections.

With *reqType* = setParam the command can be used to change the *htmr* (hold off timer) of the connection identified by the *connID* field. The new *htmr* is specified as a spec block parameter for a 1+1 connection. Changing the *htmr* is allowed for connections of type *connControl* = ONNS only. Parameter applies to 1+1 protected and permanent 1+1 protected connections.

ED-PATH-PROT to modify the protectionClass (reqType=setProt)

The **ED-PATH-PROT** command is modify the *protectionClass* of an ONNS connection.

With *reqType* = setProt the command is applicable for connections with *connControl* = ONNS (e.g. ONNS connections initiated by Management Systems).

With *reqType* = setProt the command is not applicable for connections with *connControl* = EXT (e.g. ONNS connections initiated by UNI).

With *reqType* = setProt the command can be used to change the *protectionClass* of the connection identified in the *connID* field. The new *protectionClass* is specified as the specBlock parameter *connMode*. A new *rstrPriority* can be specified at the same time.

With *reqType* = setProt the command can be used to change the reversion mode of the connection identified in the *connID* field to a 1+1 connection. The new reversion mode is specified as the specBlock parameter *connMode*. A new *wtr, htmr* can be specified at the same time.

When the *reqType* = setProt is used with the **ED-PATH-PROT** command, only explicit node list constraint for a second new path is allowed, for the exclude lists ONNS shall use the existing constraints.

The *reqType* = *setProt* function allows the user to change the conn mode from

- UNPROT to auto reroute NONREVERTIVE
- UNPROT to auto reroute REVERTIVE
- UNPROT to N1PLUS1NR
- UNPROT to N1PLUS1RV
- auto reroute NONREVERTIVE to UNPROT
- auto reroute NONREVERTIVE to auto reroute REVERTIVE
- auto reroute REVERTIVE to UNPROT
- auto reroute REVERTIVE to auto reroute NONREVERTIVE
- N1PLUS1NR to UNPROT
- N1PLUS1NR to N1PLUS1NR
- N1PLUS1NR to N1PLUS1RV
- N1PLUS1RV to UNPROT
- N1PLUS1RV to N1PLUS1NR
- N1PLUS1RV to N1PLUS1RV

ONNS chooses the active path and deletes the failed or inactive path in all cases where the new protection type is UNPROT.

The modification of the conn mode from N1PLUS1NR to N1PLUS1NR and N1PLUS1RV to N1PLUS1RV is allowed because ONNS has no knowledge about the used reversion mode of the according path protection group.

The modification of the conn mode from UNPROT to auto reroute REVERTIVE shall support the simultaneous setting of switchBack to AUTO or MAN for the resulting AUTORV connection. If no value of switchBack is specified, AUTO shall be configured.

The modification of the conn mode from auto reroute NONREVERTIVE to auto reroute REVERTIVE shall support the simultaneous setting of switchBack to AUTO or MAN for the resulting AUTORV connection.

If no value of switchBack is specified, AUTO shall be configured.

The *reqType* = setProt function allows the user to change the conn mode from

- UNPROT to P1PLUS1NRNR
- UNPROT to P1PLUS1NRRV
- P1PLUS1NRNR to UNPROT
- P1PLUS1NRNR to P1PLUS1NRRV
- P1PLUS1NRNR to P1PLUS1NRNR
- P1PLUS1NRNR to N1PLUS1NR
- P1PLUS1NRNR to N1PLUS1RV
- P1PLUS1NRRV to UNPROT
- P1PLUS1NRRV to P1PLUS1NRNR
- P1PLUS1NRRV to P1PLUS1NRRV
- P1PLUS1NRRV to N1PLUS1NR
- P1PLUS1NRRV to N1PLUS1RV
- N1PLUS1NR to P1PLUS1NRNR
- N1PLUS1NR to P1PLUS1NRRV
- N1PLUS1RV to P1PLUS1NRNR
- N1PLUS1RV to P1PLUS1NRRV.

The modification of the *connMode* from P1PLUS1NRNR to P1PLUS1NRNR and P1PLUS1NRRV to P1PLUS1NRRV is allowed because ONNS has no knowledge about the used reversion mode of the according path protection group. The modification of the *connMode* from P1PLUS1NRNR or P1PLUS1NRRV to N1PLUS1NR or N1PLUS1RV is rejected if the value of the disjointness parameter is “MAXIMAL”.

Modification of the protection type

For the modification of the protection type of a connection, specific information needs to be provided.

There are four cases taken into account:

1. Conversion from Unprotected Connections to Restoration Protected Connections
2. Conversion from Unprotected Connections to SNCP/UPSR Protected Connections
3. Conversion from Restoration Protected Connections to Unprotected Connections
4. Conversion from SNCP/UPSR Protected Connections to Unprotected Connections

For any of these modifications *reqType* = SetProt and only a subset of the input parameter apply to each case. SNCP/UPSR protected connections are 1+1 protected as also permanent 1+1 protected connections.

The following table summarizes the possible combinations for the cases 1) and 2):

Parameter Name	**Unprotected Connection**	**Unprotected Connection**
	to	**to**
	Restoration Protected Connection	**SNCP/UPSR Protected Connection**
nodeListUsage	no	yes (optional) (like nodeListUsage2 in ENT-PATH)
nodeList	no	yes (optional) (like nodeList2 in ENT-PATH)
rstrPriority (specBlock)	yes (optional)	no
connMode (specBlock)	yes (mandatory)	yes (mandatory)
disjointness (specBlock)	yes (optional)	yes (optional)
wtr (specBlock)	no	yes (optional)
htmr (specBlock)	no	yes (optional)
switchBack (specBlock)	yes (optional)	no

The following table summarizes the possible combinations for the cases 3) and 4):

Parameter Name	**Restoration Protected Connection**	**SNCP/UPSR Protected Connection**
	to	**to**
	Unprotected Connection	**Unprotected Connection**
nodeListUsage	no	no
nodeList	no	no

Parameter Name	Restoration Protected Connection	SNCP/UPSR Protected Connection
	to	to
	Unprotected Connection	Unprotected Connection
rstrPriority (specBlock)	no	no
connMode (specBlock)	yes (mandatory)	yes (mandatory)
disjointness (specBlock)	no	no
wtr (specBlock)	no	no
htmr (specBlock)	no	no
switchBack (specBlock)	no	no

Legend to the tables:

no	parameter not applicable for this modification
yes	parameter applicable or necessary for this modification
optional	if parameter is not specified the DEFAULT value will be taken
mandatory	parameter must be given

Modification of SNCP/UPSR Protected connections

For the modification of SNCP/UPSR Protected connections between normal SNCP/UPSR protected and permanent SNCP/UPSR protected, specific information needs to be provided.

There are two cases taken into account:

1. Conversion from normal SNCP/UPSR Protected Connections to Permanent SNCP/UPSR Protected Connections
2. Conversion from permanent SNCP/UPSR Protected Connections to Normal SNCP/UPSR Protected Connections

For any of these modifications *reqType* = SetProt and only a subset of the input parameter apply to each case.

The following table summarizes the possible combinations for the cases 1) and 2):

Parameter Name	Normal SNCP/UPSR Protected Connection to Permanent SNCP/UPSR Protected Connection	Permanent SNCP/UPSR Protected Connection to Normal SNCP/UPSR Protected Connection

Parameter Name	Normal SNCP/UPSR Protected Connection to Permanent SNCP/UPSR Protected Connection	Permanent SNCP/UPSR Protected Connection to Normal SNCP/UPSR Protected Connection
nodeListUsage	no	no
nodeList	no	no
rstrPriority (specBlock)	no	no
connMode (specBlock)	yes (mandatory)	yes (mandatory)
disjointness (specBlock)	no	no
wtr (specBlock)	yes (optional)	yes (optional)
htmr (specBlock)	yes (optional)	yes (optional)
switchBack (specBlock)	no	no

Legend to the tables:

no	parameter not applicable for this modification
yes	parameter applicable or necessary for this modification
optional	if parameter is not specified the DEFAULT value will be taken
mandatory	parameter must be given

Modification of Permanent SNCP/UPSR Protected connections

For the modification of Permanent SNCP/UPSR Protected connections to unprotected and vice versa, specific information needs to be provided.

There are two cases taken into account:

1. Conversion from unprotected Connections to permanent SNCP/UPSR protected connections
2. Conversion from permanent SNCP/UPSR protected connections to unprotected connections

For any of these modifications *reqType* = SetProt and only a subset of the input parameter apply to each case.

The following table summaries the possible combinations for the cases 1) and 2):

Parameter Name	Unprotected Connection to Permanent SNCP/UPSR Protected Connection	Permanent SNCP/UPSR Protected Connection to Unprotected Connection
nodeListUsage	yes (optional) (like nodeListUsage2 in ENT-PATH)	no
nodeList	yes (optional) (like nodeList2 in ENT-PATH)	no
rstrPriority (specBlock)	no	no
connMode (specBlock)	yes (mandatory)	yes (mandatory)
disjointness (specBlock)	no	no
wtr (specBlock)	yes (optional)	no
htmr (specBlock)	yes (optional)	no
switchBack (specBlock)	no	no

Legend to the tables:

no	parameter not applicable for this modification
yes	parameter applicable or necessary for this modification
optional	if parameter is not specified the DEFAULT value will be taken
mandatory	parameter must be given

REPT PTHCHG is generated

The **ED-PATH-PROT** command shall generate a REPT-PTHCHG message.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No

Input format

Command syntax: **ED-PATH-PROT**:*srcTid:srcAid:ctag::[srvID]:reqType, [connID]:::[nodeListUsage1, nodeList1]:[,linkIncludeID][:specBlock];*

Input parameters

Command parameters

1. **srcTid**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique Tid used by Host NE. See also the definition of tid in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed
   Range: The srcTid parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier srcAid of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn
3. **ctag**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages. See also the definition of ctag in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type alphanumeric string
   Range: The ctag parameter is a character strings of up to 6 ASCII characters.The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.
4. **srvID**
   The service identifier (srvID) is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   Type Integer
   Range: 1.. 20,000,000.
   Only one *srvID* accepted. The value ALL is not supported.
5. **reqType** – Request Type
   This parameter indicates the operation that is to be performed on the connection specified.
   Type: Alphanumeric value set.

Range:

- SETPARAM: modifies the value of *rstrPriority*, *wtr*, *htmr*, or *switchBack* of the connection to the value specified in the spec block.
- SETPROT: converts the connection specified by its *connID* to a new protection type specified by the *connMode* parameter in the spec block.

Default: None.

6. **`connID`** – Connection ID for an existing ONNS optical path connection service
This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and Ingress Tributary# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
Type: Character string of length up to 39.
Range: *ALL* – special value to retrieve all connection records
Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:
  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
- Index: Numeric string 0-4294967295

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-1-13-15-1&271584

Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with

- nodeID: four dot separated numeric strings: each string 0..255;
- Tributary AID: five dash separated numeric strings representing:
  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- default Tributary AID: five dash separated numeric strings with:
  Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##.
- Index: Numeric string 0-4294967295

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

7. **`nodeListUsage`** – Node List Usage
   This parameter can be used for a explicit node list (like nodeListUsage2 of ENT-PATH) for the new second path by modifying a protection type to a 1+1 connection.
   For connection setup NN used/uses the list of nodes included in the nodeList1 array block in a manner specified by the *nodeListUsage1* variable.
   Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command.
   An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent nodeID will be ignored in EXCLUDE nodeList1 during path setup. A non-existent nodeID will cause setup failure in EXPLICIT nodeList1 during path setup
   Type: Value set.
   Range: EXPLICIT: *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.); EXCLUDE: nodes in the *nodeList1* must all be excluded from the path created. In the case when the protectionClass is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the nodeList1 below also apply to the second or restoration path.

8. **`nodeList`** – Node List
   This parameter can be used for a explicit node list (like nodeList2 of ENT-PATH) for the new second path by modifying a protection type to a 1+1 connection.
   When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidX])-
   When nodeListUsage1=EXCLUDE: List of nodes that must all be excluded from both paths. The nodeID values are shown in the from: nid1([&nid2& …&nidX])
   If the preceding required nodeListUsage1 field is not supplied, this field must not be present. If user specifies empty EXCLUDE nodeList1, the connection should use indirect path from source node to destination node (i.e., at least one intermediate node is needed). If user specifies empty EXPLICIT nodeList1, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).

Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0..255.
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
Range nid1([&nisX])* or empty list; where X <= 25, when nodeListUsage1=EXPLICIT; where X <= 25, when nodeListUsage1=EXCLUSIVE.

9. **`linkIncludeID`** – VPN Identifier
A connection constraint link property such as linkIncludeID that must be valid to include the link in the connection path(s) to be created. linkIncludeID is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
Type Integer
Range: 1 … 65535 0 or an empty field indicates no VPN constraint is applied.
Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.

Spec block parameters

1. **`connAlias`** – connection alias
This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
Type: Character string up to length 83
Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.
2. **`connMode`** – Connection Mode
This parameter determines the new connection mode.
Type: Alphanumeric value set.
Range:
   - UNPROT: converts to an unprotected connection
   - N1PLUS1NR: converts to a network 1+1 protected non-revertive connection.
   - N1PLUS1RV: converts to a network 1+1 protected revertive connection.
   - NONREVERTIVE : converts to an auto-reroute non-revertive connection
   - REVERTIVE: converts to an auto-reroute revertive connection
   NOTE: The modes related to Y connections are not allowed with the **`ED-PATH-PROT`** command.
   - P1PLUS1NRNR: converts to a permanent 1+1 protected connection with non-revertive path restoration and non-revertive 1+1 protection
   - P1PLUS1NRRV: converts to a permanent 1+1 protected connection with non-revertive path restoration and revertive 1+1 protection

3. **`disjointness`**

   The parameter applies to new path only when the *reqType* requires to set up a new path such as in modifying a protection type from unprotected to a auto-reroute or Network 1+1 connection.

   The parameter does not apply to a permanent 1+1 protected connection.

   The disjointness determines the path calculation logic.

   Type: Alphanumeric value set.

   Range:

   - COMPLETE fully node and SRLG disjoint path
   - FULLYSRLG maximal node and fully SRLG disjoint path
   - MAXIMAL maximal node disjoint path.

   Maximal SRLG disjoint path calculation is not supported by the current path calculation algorithms.

   Impact on disjointness parameter values when protClass is set to N1PLUS1 or YCONN

   When *protClass* is set to N1PLUS1 or YCONN, only the *disjointness* parameter values COMPLETE and FULLYSRLG is allowed.

   Impact on disjointness parameter values when protClass is set to P1PLUS1NR

   When *protClass* is set to P1PLUS1NR, only the *disjointness* parameter values COMPLETE and FULLYSRLG is allowed.

   Default:

   - COMPLETE: for *protectionClass* N1PLUS1 and YCONN
   - MAXIMAL: for *protectionClass* AUTONR and AUTORV

     The parameter is only applicable when *protectionClass* != UNPROT.
   - COMPLETE: for *protectionClass* P1PLUS1NR

4. **`htmr`** – Hold Off Timer

   This parameter applies to 1+1 protected and permanent 1+1 protected connections.

   This parameter specifies the value of the hold-off timer for 1+1 protection switching.

   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.

   Type: numeric string [ms]

   Range:

   For high order path protection groups, the range is:

   - 0 .. 99 ms in 1 ms steps and
   - 100 .. 10000 ms in100 ms steps

For low order path protection groups, the range is:

- 0, 25, 50, 75 ms (thus in steps of 25 ms) and
- 100 .. 10000 ms in 100 ms steps.

The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
The path protection groups at the A and Z nodes of the connection are using the already provisioned values, if the parameter is not present in the spec_block.
Default: 0

5. **`switchBack`** – revertive restoration path switch back
   This parameter describes how a revertive restoration path is switched back to the original path.
   Type: Alphanumeric value set.
   Range:
   - AUTO: path is automatically switched back after wait to revert time.
   - MAN: path has to be manually switched back triggered by the TL1 command **`ED-PATH-ROLL`**.

   Default: AUTO.
6. **`rstrPriority`** – Restoration Priority
   Parameter applies to new path only when the *reqType* requires to set up a new path such as in modifying a protection type to a autoreroute connection.
   This parameter describes the restoration priority of the connection.
   Type: Integer
   Range: 0 (default) – 7 (lower values represent higher priority).
7. **`wtr`** – Wait to Restore Timer
   Parameter applies to a permanent 1+1 protected connection with revertive SNCP/UPSR protection.
   This parameter specifies the value of the wait-to-restore time for revertive 1+1 protection switching.
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
   Type: numeric string [minute].
   Range: 0 – 60.
   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
   Default: 0

Output format

If the **`ED-PATH-PROT`** request completes successfully, the following normal completion response format is returned:

```
   srcTid date time
M  ctag COMPLD
```

```
   ″ConnID=connID,ConnStatus=connStatus,pdir,,bandwidth,
   ConnAlias=connAlias,
   ,SrvID=srvID,,,,,specBlock,,
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
   PathID=pathID1,State=pathState1,
   NodeList=pNodeList1[:
   PathID=pathID2,State=pathState2,
   NodeList=pNodeList2]″
;
```

Partial system response format

The **ED-PATH-PROT** shall response with the completion code PRTL if the path protection group parameters (reversion mode, wait to restore timer or hold off timer) setup/modified with **ED-PATH-PROT** request are not successfully provisioned. The response shall contain the normal system response format to **ED-PATH-PROT** and additionally a failure information about the missing path protection group parameter provisioning:

```
   srcTid date time
M  ctag PRTL
   ″ConnID=connID,ConnStatus=connStatus,pdir,,bandwidth,
   ConnAlias=connAlias,
   ,SrvID=srvID,,,,,specBlock,,
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1:
   PathID=pathID2,State=pst2&sst2&tst2,
   NodeList=pNodeList2″

   ″E150::\″Status, Path Protection Group parameters Not Provisioned\″,″
;
```

Output parameters

The details of the output parameters *srcTid*, *ctag*, reqType, *nodeListUsage1*, *nodeList1* and their syntax and allowed values are included in the Input parameters section of the **ED-PATH** command.

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. **`date`**
   This parameter is set to the date when the output message is generate: YY-MM-DD where YY=00-99, MM=01-12, DD=01-31.

3. **`time`**
   This parameter is set to the time when the output message is generated: HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.

4. **`ctag`** – Correlation Tag
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

5. **`completionCode`**
   Standard TL1 completion code result.
   Type: Value set.

   Range:

   - COMPLD: for success (default)
   - PRTL: for partial success / failure, see section ERROR RESPONSES for details
   - DENY: in case of errors, see section ERROR RESPONSES for details

6. **`connID`** – Connection ID
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and IngressPort# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.

   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
   - Index: Numeric string 0-4294967295.

   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-1-13-15-1&1271584.

Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- defaultTributary AID: five dash separated numeric strings with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##.
- Index: Numeric string 0-4294967295.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

7. **`connStatus`** – Connection Status

   ConnStatus = pst&sst(&tst) Connection status is the overall status of the connection connID which may be different than the status of an individual *pathID* in that connection.

   connStatus is composed out of

   - pst (PrimaryState)
   - sst (SecondaryState)
   - tst (TertiaryState, optional).

   Type: up to three &-separated value sets.

   Range *pst* values:

   - CON: Connected
   - DLT: Deleted
   - NOCON: Not Connected

- MAINT: Maintenence Mode
- UNKNOWN: Conn state unknown.

Range *sst* & *tst* values

- OK: Conn good
- FAILED: Conn failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- PREMPT: Path pre-empted
- SETUPFL: Connection SetUp Failed
- FL-NVM: Not saved in NVM
- WTORV: Waiting to revert path
- EPRTFL: EDGE Port Failed
- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection)
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection)

Range *tst* values

- DEGRADED: Non-revertive restoration was not possible if the worker or protection path of a permanent 1+1 protected connection fails

8. **`pdir`** – Path Direction Topology
   Uni-direction A-Z (1WAY) or Bi-direction – (2WAY).
   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default), 2WAY.

9. **`bandwidth`** – Requested bandwidth/signalRate
   Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
   Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
10. **`connAlias`** – connection Alias
    This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
    Type: Character string up to length 83.
    Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.
11. **`srvID`** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.
12. **`srcNID`** – Source nodeID
    ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
    Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
13. **`srcPortID`** – Source PortID
    ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
    Value: dash-separated list of 5 numeric strings
    Example: 1-1-3-1-4
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups.
14. **`destNID`** – Destination nodeID
    ONNS destination nodeID. Together with the following *destPortID* field defines the associated Egress *connPoint* that connects the destination node to the edge/gateway switch or client router.
    Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

15. **`destPortID`** – Destination PortID
ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups.

16. **`pathID1`** – Path ID 1
The PathID of the 1st path that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255) (Index: Numeric string 0-65535).
The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111

17. **`pathState1`** – Path State 1
Path State1 = *pst*& *sst*[*&tst*]. pathState1 represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.
PathState1 is composed out of *pst* (PrimaryState), *sst* (SecondaryState) and *tst* (TertiaryState, optional) Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets.
Range for *pst* values:
    - CON: Connected
    - NOCON: Not Connected
    - DLT: Deleted
    - MAINT: Maintenance mode.

    Range *sst* & *tst* values:
    - OK: Conn good
    - FAILED: PATH: failed
    - SETUP: Conn SetUp occurring
    - TRDOWN: Conn teardown occurring
    - NO-IN: Conn failed, signal in bad
    - RSTOR: path restore occurring
    - RSTRD: Conn restored
    - RSTRFL: Conn restore failed
    - DLT-FL: Delete operation Failed
    - UNKNOWN: Conn state unknown

- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.
Range for *pst* values:
RSVD – Reserved, for paths associated to protecting connections within an M:N protection group, while not protecting a working traffic.

18. **`pNodeList1`** – Path Node List 1
The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

19. **`pathID2`** – Path ID 2
The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255), (Index: Numeric string 0-65535).
The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111
This are an optional output parameter in the **`ED-PATH-PROT`** command reply. Used for the second path of a N1PLUS1 and P1PLUS1NR connection.

20. **`pathState2`** – Path State 2
pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand ("&"). Individual status values are 7 characters or less.

Type: up to three &-separated value sets

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values

- OK: Conn good
- FAILED: PATH failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed

- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.
This are an optional output parameter in the **ED-PATH-PROT** command reply. Used for the second path of a N1PLUS1 and P1PLUS1NR connection.

21. **pNodeList2** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
This are an optional output parameter in the **ED-PATH-PROT** command reply. Used for the second path of a N1PLUS1 and P1PLUS1NR connection.

Spec block parameters

1. **connMode** – Connection Mode
The parameter specifies whether revertive or non-revertive mode is used as operation mode of the auto-reroute or 1+1 protected connection.
This parameter can be used with the ENT-PATH-AUTO and ENT-PATH-N1PLUS1 command. This parameter can be used with the ENT-PATH-P1PLUS1NR command to specify, whether revertive or non-revertive mode is used as operation mode of the 1+1 protection of the permanent 1+1 protected connection.
Type: Alphanumeric value set.
Range: NONREVERTIVE (default), .REVERTIVE
2. **disjointness**
The actual level of disjointness of the calculated and established connection. The parameter is not filled when protectionClass is UNPROT.
Type: Alphanumeric value set.
Range:
   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.

   This are an optional output parameter in the **ED-PATH** command reply.
3. **rstrPriority** – Restoration Priority
This parameter describes the restoration priority of the connection.
Range: 0 (default) – 7. Lower values represent higher priority.

4. **`htmr`** – Hold Off Timer
   This parameter applies to 1+1 protected and permanent 1+1 protected connections.
   This parameter specifies the value of the hold-off timer for 1+1 protection switching.
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms]
   Range:
   For high order path protection groups, the range is:
   - 0 .. 99 ms in 1 ms steps and
   - 100 .. 10000 ms in 100 ms steps

   For low order path protection groups, the range is:
   - 0, 25, 50, 75 ms (thus in steps of 25 ms) and
   - 100 .. 10000 ms in 100 ms steps.

   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
   Default: 0
5. **`wtr`** – Wait to Restore Timer
   Parameter applies to a permanent 1+1 protected connection with revertive SNCP/UPSR protection.
   This parameter specifies the value of the wait-to-restore time for revertive 1+1 protection switching.
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
   Type: numeric string [minute].
   Range: 0 – 60.
   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
   Default: 0
6. **`switchBack`** – revertive restoration path switch back
   This parameter describes how a revertive restoration path is switched back to the original path.
   Type: Alphanumeric value set.
   Range:
   - AUTO: path is automatically switched back after wait to revert time.
   - MAN: path has to be manually switched back triggered by the TL1 command **`ED-PATH-ROLL`**.

   Default: AUTO.

Example input/output

Input/output example with a protectionClass change

The following shows an **ED-PATH-PROT** example with a protection class change using SETPROT:

```
ED-PATH-PROT:LT-ONNS-256:snn:123456::12354:setprot,135.13.196.23&1-5-13-4-
      1&1::::rstrPriority=5,connMode=NONREVERTIVE;

   LT-ONNS-256 2002-01-01 08:00:00
M  123456 COMPLD
   ″ConnID=135.14.196.23&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,
   ,SrvID=12354,,,,,ConnMode=NONREVERTIVE&RstrPriority=5&
Disjointness=COMPLETE,,
   SrcNID=135.14.196.23,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.14.196.23-43516,State=CON&OK,
   NodeList=135.14.196.23&135.13.42.3&135.14.242.133&135.13.42.15″
;
```

Input/output example with a parameter change

The following shows an **ED-PATH-PROT** output example with a parameter change using SETPARAM:

```
ED-PATH-PROT:LT-ONNS-256:snn:123456::12345:setparam,135.13.196.23&1-5-13-4-
      1&1::::rstrPriority=5;

   LT-ONNS-256 2002-01-01 08:00:00
M  123456 COMPLD
   ″ConnID=135.14.196.23&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,
   ,SrvID=12345,,,,,ConnMode=REVERTIVE&RstrPriority=5&Disjointness=COMPLETE,,
   SrcNID=135.14.196.23,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.14.196.23-43516,State=CON&OK,
   NodeList=135.14.196.23&135.13.42.3&135.14.242.133&135.13.42.15″
;
```

Partial ED-PATH-PROT system response when setting path protection group parameters

If the **ED-PATH-PROT** request partial completes, the following response is returned:

```
ED-PATH-PROT: LT-ONNS-256:snn: 123456::12354:setParam,135.13.196.23&1-5-13-4-
      1&1::::htmr=100;

   LT-ONNS-256 2002-01-01 08:00:00
M  123456 PRTL
   ″ConnID=135.14.196.23&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,,STS48,
```

```
   ConnAlias=AmericanExp-NY-BOS-ckt,
   ,SrvID=12345,,,,,RstrPriority=0&Disjointness=COMPLETE,,
   SrcNID=135.14.196.23,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.14.196.23-43516,State=CON&OK,
   NodeList=135.14.196.23&135.13.42.3&135.14.242.133&135.13.42.15:
   PathID=135.14.196.23-43517,State=CON&OK,
   NodeList=135.13.22.3&135.13.44.93&135.13.196.123"

   "E150::\"Status, Path Protection Group parameters Not Provisioned\","
;
```

Error responses

Validation rule: Insufficient network resources for alternate route

If network resources for an alternate route of a connection to be changed to a restoration protected connection are insufficient, then the following error response is returned:

Error code:
`E004 – srvID – InSufficient Network Resources for Alternate Path`

The following list identifies conditions that will cause similar error responses:

Error code: `E012 – SXCF – Status, XConn Failure caused path setup failure`

Error code:
`E013 – SSCF – Status, Signaling Communication link Failure (or abort) caused path operation failure`

Error code: `E014 – SXCT – Status, XConn Timeout`

Error code: `E016 – SHTO – Status, Host interface TimeOut`

Error code: `E017 – SSTO – Status, Signaling Timeout`

Error code:
`E022 – SPSF – Status, Path Setup Failure caused by signal transmission failure at egress node`

E016 and E022 are only sent in notifications, not in command replies

Validation rule: Maximum delay errors on calculated path

The maximum transport delay exceeded error response is shown below.

Error code: `E020 – Maximum Transport Delay Exceeded (by calculated path)`

Validation rule: ED-PATH-PROT request denied - Parameter invalid

The request parameter invalid error response is shown below:

Error code: `E030 – Syntax, Parameter Data Invalid, invalid_parameter`

The following list identifies conditions that will cause similar error responses:

Error code: `E032 – Input, Source nodeID is invalid`

Error code: `E033 – Input, Node List has Invalid / unknown ID`

Error code: `E035 -ICAE – Input, ConnAlias Already Exists`

Error code: `E037 – INCS – Input, Neither ConnID nor ConnAlias Specified`

The following error response will be raised in case connection alias and connection ID specified in the command do not match:

Error code: `E038 – IICS – Input, Inconsistent connAlias Specified`

Validation rule: ED-PATH-PROT syntax, invalid degree of disjointness value

When an invalid degree of disjointness value for protClass N1PLUS1 or YCONN is used, then the following error response is returned:

Error code:

```
E520 – IIDD – Input, Invalid Degree of Disjointness for protection type
N1PLUS1 or YCONN
```

Validation rule: ED-PATH-PROT Path Protection Group Parameter transfer failed

Error code:

```
E150 – Status, Path Protection Group parameters Not Provisioned
```

The following list identifies conditions that will cause similar error responses:

Error code: `E335 – SCOF Error – Status, Connection Operation Failed`

Error code:

```
E336 – SCTP Error – Status, Invalid command for a Connection with Two
established Paths
```

The following list identifies conditions that will cause similar error responses:

Error code:

```
E351 – SPOA – Status, Path Operation not Allowed on connection in
Restoration queue
```

Validation rule: Port related problems

If the NE receives an ED-PATH-MODIFY command where the *portID* parameter value is not valid, then one of the following error responses is returned:

Error code:

```
E082 – Status, NNI connPoint / port Unavailable ID=<ASTN node ID>
&<ConnPtID>
```

The following list identifies conditions that will cause similar error responses:

Error code:
`E083 – SCIU – Status, NNI connPoint in Use, XConn or reservation exists, ID=<ASTN node ID>&<ConnPtID>`

Error code:
`E161 – CFUT – XConn Connect Failure, Unexpected toplogy or related cross connect, ID=<ASTN node ID>&<ConnPtID>`

Error code:
`E162 – CFXE – XConn Connect Failure, Cross connect already exists, ID=<ASTN node ID>&<ConnPtID>`

Some of these errors can occur in the process of path setup and are similar to those found in ENT-PATH. The following condition will cause a similar error response:

Error code: `E279 – SNRE – Status NVM Record Exhausted`

This failure condition is only valid until a variable NVM allocation scheme is implemented, that does not have any restrictions wrt the length of connection records to be stored.

Validation rule: protectionClass is not allowed

If the NE receives an **`ED-PATH-PROT`** or **`ED-PATH-ROLL`** command or a connection of *protectionClass* MBYNWRK or MBYNPRT, then the following error response is returned:

Error code:
`SIPC – E091::Status, Invalid command for protection class MBYNWRK or MBYNPRT`

Validation rule: Protection modification for External Connections

If the network element receives an **`ED-PATH-PROT`** command for an externally controlled switched connection (*connControl* = EXT, that is, setup as part of a network connection, for example, via UNI), then the following error response is returned:

```
   srcTid date time
M  ctag DENY
   SIEC
   ″E092::\″Status, Invalid command for Externally Controlled Connections\″
;
```

The following list identifies further conditions that will cause error responses:

- Error code:
  `E360 – SPMF – Status, Path Modify Failed due to invalid protection type change`

Validation rule: Switch Back behavior modification

The following error response is returned, if a auto reroute revertive connection in a wrong state (different from CON&OK, CON&NO-IN) is modified in its switch back behavior:

Error code:

`E282 – SCSI Error – Status, Conn State Invalid for this Operation`

The following list identifies error conditions wrt. changing protection type of adaptive ONNS connections:

- Error code:
  `E170 – IIPA – Input, Invalid Protection type for Adaptive ONNS connection`

The following error response shall be raised in case the protection type of the ONNS connection cannot be changed due to a forced cross connection loopback existing on the EDGE tributary in the A or Z node:

Error code:

`E284 – SFCL – Status, Forced Cross Connection Loopback existing on EDGE tributary`

Note that this restriction only affects protection type changes requiring the setup of a path associated to the connection. Protection type changes not requiring such an action are not affected.

Related TL1 messages

- “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698)
- “ENT-PATH: Create ONNS path connection” (p. 3-1701)
- “DLT-PATH: Delete ONNS path connection” (p. 3-222)
- “RTRV-NNCNFG: Retrieve ONNS configuration data” (p. 3-2555)
- “ED-NNCNFG: Modify ONNS system configuration” (p. 3-450)
- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “RTRV-NNDR: Retrieve ONNS neighbor nodal data repository” (p. 3-2561)
- “RTRV-NTDR: Retrieve ONNS network topology data repository” (p. 3-2588)
- “REPT-NNCHG notification” (p. 3-2137)
- “REPT-NNEVT notification” (p. 3-2144)
- “ED-PATH: Add or modify alias of ONNS connection” (p. 3-739)
- “ED-PATH-ROLL: Modify ONNS connection route” (p. 3-781)

ED-PATH-ROLL: Modify ONNS connection route

Overview

The **ED-PATH-ROLL** command is used to modify the route of a connection between the two client ports via a bridge and roll.

ED-PATH-ROLL is used to modify a path location

The **ED-PATH-ROLL** command is used tp modify the existing connection service based on its *connID* by rolling an existing path or connection to a new path. The *reqType* parameter specifies what type of path operation(s) occurs on the connection in the context of a roll command.

With *reqType* = *rollPath* it is possible to modify a connection in a way to construct a new path bridged from the source port to the destination port that is setup using new constraints that are included in the delay, *srlgExcludeList*, *linkIncludeID*, and *nodeList* fields. If no new constraints are given, ONNS uses no user constraints and sets up the path using the least cost criteria. When the new path is stable, the old path is removed to complete the ONNS roll process. A new *rstrPriority* can be specified at the same time.

Roll of a 1+1 connection is not supported directly, the connection must be converted to an unprotected connection and then rolled. Then the connection can be modified back to a network 1+1 connection.

The **ED-PATH-ROLL** command is not applicable to NN connections of protectionClass MBYNWRK or MBYNPRT.

ED-PATH-ROLL is used to manually switch back a revertive restoration connection

The **ED-PATH-ROLL** command is used for manually switch back a revertive restoration connection based on its *connID* by switching back from the restoration path to the original path The *reqType* parameter specifies the type of path operation(s) occurs on the connection in the context of a roll command.

With *reqType* = SWBK it is possible to switch from the restoration path to the original path of a revertive restoration connection if the following criteria are fulfilled:

- the connection was setup with the switch back criteria “MAN”
- the connection is in the state CON&OK&WTORV

The **ED-PATH-ROLL** command is applicable for connections with *connControl* = ONNS (e.g. ONNS connections initiated by Management Systems).

The **ED-PATH-ROLL** command is applicable for connections with *connControl* = SC (e.g. ONNS connections initiated by UNI).

The **ED-PATH-ROLL** command is applicable for connections with *connControl* = EXT (e.g. ONNS connections initiated by UNI).

User defined connection alias used as an alternate key

Instead of the connID, the user may optionally use the connAlias as an alternate key. This applies to modifiers such as ED-PATH-PROT and ED-PATH-ROLL. The **ED-PATH** command could have either or both specified. If both are specified by the user, then *ConnID* id used as the key.

Either the *connID* or the *connAlias* must be present in a command. The *connAlias* is returned as an output parameter. If it is not defined, a NULL is returned.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No

Input format

Command syntax: **ED-PATH-ROLL:***srcTid:[srcAid]:ctag::[srvID]:reqType, [connID]:[maxDelay]:[nodeListUsage,nodeList] [:srlgExcludeList][,linkIncludeID][:specBlock];*

Input parameters

Command parameters

1. **srcTid**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique Tid used by Host NE. See also the definition of tid in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed
   Range: The srcTid parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier srcAid of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: *snn*
3. **ctag**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages. See also the definition of ctag in "RTRV-HDR: Retrieve header" (p. 3-2417).

Type alphanumeric string
Range: The ctag parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.

4. **`srvID`**
   The service identifier (srvID) is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   Type Integer
   Range: 1.. 20,000,000. Only one srvID accepted. The value ALL is not supported.
5. **`reqType`** – Request Type
   This parameter indicates the operation that is to be performed on the connection specified.
   Type: Alphanumeric value set.
   Range:
   - ROLLPATH: used to add a bridged/merge path to an existing connection and then roll to the new path.
   - SWBK: used to switch from a restoration path to the original path

   Default: n/a
6. **`connID` –** Connection ID for an existing ONNS optical path connection service
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and Ingress Tributary# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.
   Range: *ALL* – special value to retrieve all connection records
   Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with
   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:
     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-1-13-15-1&271584
   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with
   - nodeID: four dot separated numeric strings: each string 0..255;
   - Tributary AID: five dash separated numeric strings representing:

Bay/Frame #(1), Shelf #(1), Slot/Pack #(1 – 39), Port #(1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- default Tributary AID: five dash separated numeric strings with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##.
- Index: Numeric string 0-4294967295

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

7. **`maxDelay`** – Maximum transport delay

   Parameter applies to new path only when the *reqType* requires to set up a new path such as in rolling to new path.

   Maximum transport delay through all nodes in the path. Total path delay must not exceed this value.

   Range: 0, is used to select no delay limit; 1 – 32767, maximum transport delay in milliseconds.

   Default: 0.

8. **`nodeListUsage`** – Node List Usage

   This parameter can be used for both values (exclude, explicit) to roll a path.

   Parameter applies to new path only when the *reqType* requires to set up a new path such as in rolling to new path.

   For connection setup NN used/uses the list of nodes included in the nodeList1 array block in a manner specified by the nodeListUsage1 variable.

   Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command.

   An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent nodeID will be ignored in EXCLUDE nodeList1 during path setup. A non-existent nodeID will cause setup failure in EXPLICIT nodeList1 during path setup..

   Type: Value set.

   Range: EXPLICIT: *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.). EXCLUDE: nodes in the *nodeList1* must all be excluded from the path created. In the case when the protectionClass is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the nodeList1 below also apply to the second or restoration path.

9. **`nodeList`** – Node List
   This parameter can be used to roll to a new path.
   Parameter applies to new path only when the *reqType* requires to set up a new path such as in rolling to new path.
   When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidX])
   When *nodeListUsage1* = EXCLUDE: List of nodes that must all be excluded from both paths. The nodeID values are shown in the from: nid1([&nid2& …&nidX])
   If the preceding required nodeListUsage1 field is not supplied, this field must not be present. If user specifies empty EXCLUDE nodeList1, the connection should use indirect path from source node to destination node (i.e., at least one intermediate node is needed). If user specifies empty EXPLICIT nodeList1, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
   Type: &-separated list of nodeIDs A nodeID is represented as four dot separated numeric string with each numeric string between 0..255.
   Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
   Range nid1([&nisX])* or empty list; where X <= 25, when nodeListUsage1=EXPLICIT; where X <= 25, when nodeListUsage1=EXCLUSIVE.

10. **`srlgExcludeList`** – SRLG Exclude List
    Parameter applies to new path only when the reqType requires rolling to new path. List of Shared Risk Link Group identifiers that are to be used to exclude links from the connection path(s) being created. *srlgID* (or Shared_Risk_LinkGrpID) identifiers are properties of links between ONNS nodes.
    Type: &-separated list of numeric strings | empty | 0
    Example: 33&49&64
    Range: up to 25 numeric strings, each in the range 1…65535. The value 0 or an empty field indicates no constraint are applied.

11. **`linkIncludeID`** – VPN Identifier
    A connection constraint link property such as linkIncludeID that must be valid to include the link in the connection path(s) to be created. *linkIncludeID* is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
    Type Integer
    Range: 1 … 65535. 0 or an empty field indicates no VPN constraint is applied.
    Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.

Spec block parameters

1. **`connAlias`** – connection alias
   This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.
2. **`rstrPriority`** – Restoration Priority
   Parameter applies to new path only when the reqType requires to set up a new path such as in rolling to new path.
   This parameter describes the restoration priority of the connection.
   Type Integer
   Range: 0 … 7 (lower values represent higher priority)

Output format

If the **`ED-PATH-ROLL`** request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″ConnID=connID,ConnStatus=connStatus,pdir,,bandwidth,ConnAlias=connAlias,
   ,SrvID=srvID,,,,,specBlock,
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1:
   PathID=pathID2,State=pst2&sst2&tst2,
   NodeList=pNodeList2″
;
```

Output parameters

Command parameters

1. **srcTid**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **date**
   This parameter is set to the date when the output message is generate: YY-MM-DD where YY=00-99, MM=01-12, DD=01-31.
3. **time**
   This parameter is set to the time when the output message is generated: HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.
4. **ctag** – Correlation Tag
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.
5. **completionCode**
   Standard TL1 completion code result.
   Type: Value set.

   Range:

   - COMPLD: for success (default)
   - PRTL: for partial success / failure, see section ERROR RESPONSES for details
   - DENY: in case of errors, see section ERROR RESPONSES for details
6. **connID** – Connection ID
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and IngressPort# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.

   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).

- Index: Numeric string 0-4294967295.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-1-13-15-1&1271584.

Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG.
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

Range: *nodeID&defaultTributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- defaultTributary AID: five dash separated numeric strings with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##.
- Index: Numeric string 0-4294967295.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

Range: *nodeID&groupIndex* - to retrieve all connections belonging to an M:N protected connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.

7. **`connStatus`** – Connection Status

   ConnStatus = pst&sst(&tst) Connection status is the overall status of the connection *connID* which may be different than the status of an individual *pathID* in that connection.

   connStatus is composed out of

   - pst (PrimaryState)
   - sst (SecondaryState)
   - tst (TertiaryState, optional).

   Type: up to three &-separated value sets.

Range *pst* values:

- CON: Connected
- DLT: Deleted
- NOCON: Not Connected
- MAINT: Maintenence Mode
- UNKNOWN: Conn state unknown.

Range *sst* & *tst* values

- OK: Conn good
- FAILED: Conn failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- PREMPT: Path pre-empted
- SETUPFL: Connection SetUp Failed
- FL-NVM: Not saved in NVM
- WTORV: Waiting to revert path
- EPRTFL: EDGE Port Failed
- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values:

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection)
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection)

Range *tst* values:

- DEGRADED: Non-revertive restoration was not possible if the worker or protection path of a permanent 1+1 protected connection fails

8. **pdir** – Path Direction Topology
   Uni-direction A-Z (1WAY) or Bi-direction – (2WAY).
   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default), 2WAY
9. **bandwidth** – Requested bandwidth/signalRate
   Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
   Type: Value set.
   Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
10. **connAlias** – connection Alias
    This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
    Type: Character string up to length 83.
    Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.
11. **srvID** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.
12. **srcNID** – Source nodeID
    ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
    Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
13. **srcPortID** – Source PortID
    ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
    Value: dash-separated list of 5 numeric strings
    Example: 1-1-3-1-4
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups
14. **destNID** – Destination nodeID
    ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
    Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

15. **`destPortID`** – Destination PortID
    ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
    Value: dash-separated list of 5 numeric strings
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups
16. **`pathID1`** – Path ID 1
    This is the *pathID* of the rerouted path.
    The PathID of the 1st path that was created.
    Type: character string of length up to 20.
    Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255); (Index: Numeric string).
    The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
    Example: 29.13.196.203-65111.
17. **`pathState1`** – Path State 1
    This is the path state of the rerouted path.
    Path State1 = *pst*& *sst*[*&tst*]. *pathState1* represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.
    PathState1 is composed out of *pst* (PrimaryState), *sst* (SecondaryState) and *tst* (TertiaryState, optional) Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand ("&"). Individual status values are 7 characters or less.
    Type: up to three &-separated value sets.
    Range *pst* values:
    - CON: Connected
    - NOCON: Not Connected
    - DLT: Deleted
    - MAINT: Maintenance mode.

    Range *sst* & *tst* values
    - OK: Conn good
    - FAILED: PATH failed
    - SETUP: Conn SetUp occurring
    - TRDOWN: Conn teardown occurring
    - NO-IN: Conn failed, signal in bad
    - RSTOR: path restore occurring
    - RSTRD: Conn restored
    - RSTRFL: Conn restore failed
    - DLT-FL: Delete operation Failed

- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

Range *pst* values:

- RSVD: Reserved, for paths associated to protecting connections within an M:N protection group, while not protecting a working traffic.

18. **`pNodeList1`** – Path Node List 1
This is the node list of the rerouted path.
The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

19. **`pathID2`** – Path ID 2
The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255), (Index: Numeric string 0-65535).
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111
This is the path ID of the ″old″ path.

20. **`pathState2`** – Path State 2
pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.

Type: up to three &-separated value sets

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values

- OK: Conn good
- FAILED: PATHfailed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed

- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.
This is the path state of the ″old″ path.

21. **`pNodeList2`** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
This are an optional output parameter in the **`ED-PATH`** command reply.
This is the node list of the ″old″ path.

Spec block parameter

1. **`connMode`** – Connection Mode
This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the auto-reroute or 1+1 protected connection.
This parameter can be used with the ENT-PATH-AUTO and ENT-PATH-N1PLUS1 command. This parameter can be used with the ENT-PATH-P1PLUS1NR command to specify, whether revertive or non-revertive mode is used as operation mode of the 1+1 protection of the permanent 1+1 protected connection.
Type: Alphanumeric value set.
Range: NONREVERTIVE (default), REVERTIVE
2. **`disjointness`**
The actual level of disjointness of the calculated and established connection. The parameter is not filled when protectionClass is UNPROT.
Type: Alphanumeric value set.
Range:
   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.
3. **`rstrPriority`** – Restoration Priority
This parameter describes the restoration priority of the connection.
Range: 0 – 7. Lower values represent higher priority.
Default: 0.

Example input/output

The following example shows a successful execution of the **`ED-PATH-ROLL`** command to relocate a connection with new setup constraints:

```
ED-PATH-ROLL:LT-ONNS-256:snn:123459::19265478:rollPath,135.14.196.23&1-5-3-4-
      1&2:61:Exclude,135.13.42.3&135.213.242.103&135.13.198.22:65&73&83&19:RstrPriority=5;
```

```
   LT-ONNS-256 2002-09-01 08:00:00
M  123459 COMPLD
   ″ConnID=135.14.196.23&1-5-3-2-49&2,ConnStatus=CON&OK,2WAY,,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,,
   SrvID=1351465478,,,,ConnMode=NONREVERTIVE&RstrPriority=5&Disjointness=COMPLETE,,
   SrcNID=135.14.196.23,1-5-3-2-49,DestNID=135.13.196.123,1-2-5-2-1:
   PathID=135.14.196.23-43876,State=CON&OK,
   NodeList=135.14.196.23&135.13.42.3&135.14.242.133&135.13.196.123:
   PathID=135.14.196.23-54326,State=CON&SETUP,
   NodeList=135.14.196.23&135.13.42.153&135.14.242.133&135.13.196.132″
;
```

Error responses

Validation rule: Insufficient network resources for alternate route

If network resources for an alternate route of a connection to be changed to a restoration protected connection are insufficient, then the following error response is returned:

```
   srcTid 2002-06-01 08:00:00
M  ctag DENY
   ISNR
   ″E001:srvID:\″InSufficient Network Resources for connection setup\″,″
;
```

Error code:
`E001 – srvID – InSufficient Network Resources for connection setup`

The following list identifies conditions that will cause similar error responses:

Error code: `E012 – SXCF – Status, XConn Failure caused path setup failure`

Error code:
`E013 – SSCF – Status, Signaling Communication link Failure (or abort) caused path operation failure`

Error code: `E014 – SXCT – Status, XConn Timeout`

Error code: `E016 – SHTO – Status, Host interface TimeOut`

Error code: `E017 – SSTO – Status, Signaling Timeout`

Error code:
`E022 – SPSF – Status, Path Setup Failure caused by signal transmission failure at egress node`

E016 and E022 are only sent in notifications, not in command replies

Validation rule: Maximum delay errors on calculated path

The maximum transport delay exceeded error response is shown below.

Error code: `E020 – Maximum Transport Delay Exceeded (by calculated path)`

Validation rule: ED-PATH request denied - Parameter invalid

The request parameter invalid error response is shown below:

```
   srcTid 2002-06-01 08:00:00
M  ctag DENY
   SPDI
   ″E030::\″Syntax, Parameter Data Invalid, invalid_parameter\″,″
;
```

Example: The command

```
ED-PATH-ROLL:srcTid:srcAid:ctag::srvID:rollPath,connID1::
   Explicit,BadNode5ID,Node6ID:srlgExcludeList;
```

may produce this result.

The following list identifies conditions that will cause similar error responses (including E030 shown above):

- Error code: `E030 – SPDI – Syntax, Parameter Data Invalid, invalid_parameter= fieldName`
- Error code: `E032 – ISIP – Input, Source nodeID is invalid`
- Error code: `E033 – INLI – Input, Node List has Invalid / unknown ID`
- Error code: `E037 – INCS – Input, Neither ConnID nor ConnAlias Specified`

The following error response will be raised in case connection alias and connection ID specified in the command do not match:

- Error code: `E038 – IICS – Input, Inconsistent connAlias Specified`

The following error response is raised if the connection is not a restoration revertive connection or if the switch back behavior for a revertive restoration connection is set to “AUTO”:

- Error code: `E039 – ISBF – Input, Criteria for manual Switch Back not Fulfilled`

The following error response is raised if a restoration revertive connection with switch back behavior set to “MAN” is not in state CON&OK&WTORV:

- Error code: `E282 – SCSI – Status, Conn State Invalid for this Operation`

The following error response is raised in case the ONNS connection cannot be rerouted due to a forced cross-connection loopback existing on the EDGE tributary in the A or Z node:

- Error code:
  `E284 – SFCL – Status, Forced Cross Connection Loopback existing on EDGE tributary`

Related TL1 messages

- “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698)
- “ENT-PATH: Create ONNS path connection” (p. 3-1701)
- “DLT-PATH: Delete ONNS path connection” (p. 3-222)
- “RTRV-NNCNFG: Retrieve ONNS configuration data” (p. 3-2555)
- “ED-NNCNFG: Modify ONNS system configuration” (p. 3-450)
- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “RTRV-NNDR: Retrieve ONNS neighbor nodal data repository” (p. 3-2561)
- “RTRV-NTDR: Retrieve ONNS network topology data repository” (p. 3-2588)
- “REPT-NNCHG notification” (p. 3-2137)
- “REPT-NNEVT notification” (p. 3-2144)
- “ED-PATH: Add or modify alias of ONNS connection” (p. 3-739)
- “ED-PATH-PROT: Modify ONNS connection parameters or protection type” (p. 3-753)

ED-PID: Edit password identifier

The **ED-PID** command can be initiated by a user to change the user's own password on the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-PID**:*tid*::*ctag*::*old_pid*,*new_pid*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **old_pid** – Old Password ID
   This is the current *pid* of the user.
   This required parameter is the confidential password authenticator for the given *uid*. Password strings are transmitted in non-encrypted form in the **ACT-USER** command; are encrypted when stored in the NE; and are never transmitted from the NE.
   Type: character string (case-sensitive)
   Range: 6 to 10 legal characters. Valid passwords consist of at least two non-alphabetic characters with at least one special character. The special character can be one of the required non-alpha characters. For example, TEST1+ is a valid password. The first character of a password must be a letter.
   Default: For the two superusers the default password is: UNITE+01 for User ID LUC01 and UNITE+02 for User ID LUC02. For other users: not applicable.
4. **new_pid** – New Password ID
   This parameter specifies the new password ID. The *new_pid* value must differ from the *old_pid* value by at least one character.

Output format

After receiving the **ED-PID** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

```
;
```

If the password change request completes successfully during login procedure after the password expired and transitions the state of the login procedure from the "Password_Expired" state to the "Login_Active" state, the following normal completion response from the **ACT-USER** command is returned when no customer proprietary banner is provisioned:

```
   sid date time
M  ctag COMPLD
   "uid:lastdate,lasttime,attempts,systype,release,upc:spec_block"
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release release
   User Privilege Code: upc
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENT AND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */

;
```

If the password change request completes successfully during login procedure after the password expired and transitions the state of the login procedure from the "Password_Expired" state to the "Login_Active" state, the following normal completion response from the **ACT-USER** command is returned when a customer proprietary banner is provisioned:

```
   sid date time
M  ctag COMPLD
   "uid:lastdate,lasttime,attempts,systype,release,upc:spec_block"
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release release
   User Privilege Code: upc
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENT AND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */
   /*
   <LINE1>
```

```
   <LINE2>
    ...
    ...
    ...
   <LINE10>
   <LINE11>
   */

;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

When the NE receives an **ED-PID** command, the *pid* (password ID) associated with the user's login identifier will be changed on the NE provided that:

- the user is currently logged in on the NE
- the *old_pid* provided matches the current password for that login identifier
- the *new_pid* provided satisfies the password requirements of the NE.

It will be reported via the corresponding ED-PID notification – see "ED-PID notification" (p. 3-802) section – if a session is already established. In the password expiry scenario during a login procedure, the ED-PID notification is optional and may be omitted..

When the **ED-PID** command is successfully executed, the user session is not terminated and is unaffected. It will continue to use the previous values. The changes will take effect the next time the user logs in.

Example input/output

The following example shows a successful password change operation:

ED-PID:LT-UNITE-123::123456::soup3r+,mash%52;^

```
   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a successful completion response of **ED-PID** while attempting the **ACT-USER** command:

ED-PID:UNITE-123::123456::soup3r+,mash%52;

```
   UNITE-123 07-12-21 08:00:00
M  123456 COMPLD
   " nbguser1:07-12-20,09-00-00,1,WaveStar_UNITE,10.00.00,P3&M3"
   /* Alcatel-Lucent 1675 LambdaUnite MSS Release 10.00.00
```

```
   User Privilege Code: P3&M3
   ALCATEL-LUCENT - PROPRIETARY
   THIS SOFTWARE CONTAINS INFORMATION OF ALCATEL-LUCENT
   AND IS NOT TO BE DISCLOSED OR USED EXCEPT IN ACCORDANCE
   WITH APPLICABLE AGREEMENTS.
   NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.
   USE OF THIS SOFTWARE IS GOVERNED SOLELY AS EXPRESSLY
   AUTHORIZED IN THE RELEVANT AGREEMENT BETWEEN
   ALCATEL-LUCENT AND CUSTOMER.
   UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.
   */
;
```

Error responses

Validation rule: Invalid *old_pid*

If the NE receives an ED-PID command with an *old_pid* value that does not match with the password ID of the associated user login ID.

Error code: `IDNV – Input, Data Not Valid, wrong old_pid`

Validation rule: Missing or invalid or syntactically incorrect *new_pid*

If the NE receives an ED-PID command with a missing *new_pid* or the *new_pid* is same as the *old_pid* or the *new_pid* is syntactically incorrect.

Error code: `IDNV – Input, Data Not Valid, incorrect new_pid`

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

ED-PID notification

Output format

See definition of "REPT DBCHG: Report data base change" (p. 3-2118).

The ED-PID notification has the following specific syntax:

ED-PID:*uid*

Output parameters

Command parameters

1. **uid** – User ID
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 1 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.

Example output

The example shows a normal notification after an ED-PID command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-PID:nbguser1″
;
```

ED-PROTN-GRP: Edit protection group

The **ED-PROTN-GRP** command allows one to modify parameter/attributes related to an already existing protection group instance of any protection group type.

Note: Manual protection switching is done by means of commands **OPR-PROTNSW**, **RLS-PROTNSW**.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

Multiplex section/line protection group

Command syntax: **ED-PROTN-GRP**:*tid*:*aid*:*ctag*::*protype*:*spec_block;*

2-/4-fibre MS-SPRing/BLSR protection group

Command syntax: **ED-PROTN-GRP**:*tid*:*[aid]*:*ctag*::*protype,[rid]*:*spec_block*;

Path protection group

Command syntax: **ED-PROTN-GRP**:*tid*:*[aid]*:*ctag*::*protype, ,[ppgname][,aid1]*:*spec_block*;

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.

Input parameters

Command parameters (multiplex section/line protection group)

1. **tid** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port aid being a member of the group.
   Type: AID
   Range: OnePlusOne_ProtGrpAID (not ranged) or NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged).

3. **ctag** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **protype** – Protection Type
   The command parameter *protype* is used for additional identification beside the *aid*, it is not usable to change the value of protype.
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later)
   Type: Value set.
   Range: 1+1 or 1xNOPT

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **tid** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged).
3. **ctag** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **protype** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F
5. **rid** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.

Command parameters (path protection group)

1. **tid** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies a path protection group aid or an input tributary aid being a member of the group in combination with *ppgname*.
   This parameter specifies an input tributary *aid* when used in combination with parameter *aid1*.
   Option: If *ppgname* is specified, then *aid* can be omitted or used as an input tributary aid.
   For low order path protections, the *aid* cannot be omitted as the *ppgname* is not supported for those protection groups. For such protection groups always a path protection group AID has to be specified.
   The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Option: If parameters aid specifies a HO tributary on a port of port class EDGE and this tributary is combined with another HO tributary within an OUC, then ED-PROTN-GRP shall operate on all protection groups of type PATHDRI identified by these two HO tributaries, if existing. If aid specifies a HO tributary on a port of port class INNI, the command is rejected.
   Option: When parameter *aid* is used in combination with parameter *aid1*, then ED-PROTN-GRP shall operate on all protection groups of type PATHDRI having the AIDs given in parameters *aid* and *aid1* as input tributaries. If *aid* or *aid1* specifies a tributaries on a port of port class INNI, the command is rejected.
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
   The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **all** or **vall** or **usall** in various positions) are allowed.
3. **ctag** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **protype** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: PATHDRI
5. **ppgname** – Path protection group Name
   In particular follow the reference to check options and mandatory omissions associated with this parameter.
   The parameter specifies the path protection group name. Storage and retrieval of the ppgname is case sensitive, but actual use of the ppgname to determine the

corresponding protection group is case insensitive.
Option for high order path protection groups: For high order path protections, if *aid* is specified, then *ppgname* can be omitted. For low order path protections, the *ppgname* is omitted as not being supported at all. If an input value is specified for *ppgname* and none is specified for *aid*, then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified path protection group name.
If input values are specified for *aid* and *ppgname* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified *aid*, *ppgname* pair. The command will complete successfully if there is no protection group described by *aid*, *ppgname*, but no data will be returned.
The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
Type: text string
Range: 1 to 24 characters

6. **`aid1`** – Access identifier 1
   Together with parameter *aid* this parameter specifies a pair of input tributaries in order to access emulated selector bridges in one command.
   If input values are specified for *aid* and *aid1* then ED-PROTN-GRP operates on all protection groups of type PATHDRI that have these specified tributaries as inputs.If input values are specified for *aid* and *aid1*and additionally a value is specified for *ppgname*, ED-PROTN-GRP operates only on those protection groups that additionally match the ppgname valu.. If *aid1* specifies a HO tributary in a port of port class INNI, the command is rejected.
   Type: AID
   Range: STS1NumAID
   The use of wildcards is not allowed.

Spec block parameters (multiplex section/line protection group)

1. **`apspn`** – MSP-APS ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MSP-APS (Multiplex Section Protection – Automatic protection switching).
   Type: Alphanumeric string
   Range: 1-24 characters including the character "-".
   Default: DEFAULT
2. **`dcclmsprotmd`** – DCC Line/MS Protection Mode
   The parameter specifies, whether the line/MS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set 3;
   Range: ENABLE, DISABLE
   Default: DISABLE

3. **dccsrsprotmd** – DCC Sect/RS Protection Mode
   The parameter specifies, whether the section/RS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set
   Range: ENABLE, DISABLE
   Default: DISABLE
4. **wtr** – Wait To Restore Time
   The parameter specifies the "wait to restore" time used by the protection state machine operating in revertive mode. In case of MSP 1+1 optimized, the same parameter is used for the "wait to rename" time.
   Type: numeric string [minute]
   Range: 0 – 60
   Default: 5.

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

1. **ard** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value manually with ED-MAP-RING and supply the ring map information to each node on the ring for the BLSR protocol to operate. It is necessary to activate this user-provisioned information with SET-MAP-RING. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
   Type: Alphanumeric string.
   Range: ENABLE, DISABLE.
   Default: ENABLE.
2. **asmc** – Automatic Squelch Map Calculation
   If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
   Type: alhpanumeric string
   Range: ENABLE, DISABLE (default)
3. **blsrpn** – MS-SPRing-BLSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
   Type: Alphanumeric string
   Range: 1-24 characters including the character "-".
   Default: DEFAULT

4. **`newrid`** – New Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
5. **`rcamode`** – Ring Circuit Alarm Mode
   This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
6. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
   Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
   Type: numeric string [minute].
   Range: 0 – 12 in 1 minute increments and 99.
   Default: 5.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value and supply the ring map information to each node on the ring for the BLSR protocol to operate. Until all

nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.

2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: alhpanumeric string
Range: ENABLE, DISABLE (default)
3. **`blsrpn`** – MS-SPRing-BLSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
Type: Alphanumeric string.
Range: 1-24 characters including the character "-".
Default: DEFAULT.
4. **`eastwkg`** – East Worker Port
This parameter specifies the port being a member in the role of the east worker port for a 4-fibre MS-SPRing/BLSR protection group.
Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID.
Range: In case of an asymmetric ring: NOTUSED. In all other cases: SPortAID.
5. **`newrid`** – New Ring Identification
This parameter specifies the new ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the *rid* to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE
6. **`protnmode`** – Protection Switch Mode
The parameter specifies the protection switch mode. The value LOOPBACK is used for the classical ring application. The value SHORTENED is used for transmission delay sensitive applications. The value TOPEX (only R3.0 - R3.2 and later) is used for the so called transoceanic application, which allows extra traffic, improves restorability and improves delay as well as in SHORTENED mode.
In case of a ring switch … LOOPBACK performs the bridge and switch function in that nodes which are most close to the failed span. SHORTENED performs the bridge function at the add node repectively the switch function at the drop node of each ring circuit. It is an application only similar to the transoceanic protocol. TOPEX allows, compared to SHORTENED mode, additionally the traffic restoration of not affected spans. It is the transoceanic protocol application.
Consistent Protection Switch Mode Setting: The Protection Switch Mode must be set consistently on all nodes in the ring. E.g. for all UNITE NEs in a TOP+EX ring

the parameter protnmode must be set to TOPEX. In a ring with inconsistent Protection Switch Mode, there will be no protection switching. The UNITE NEs will raise an Inconsistent Ring Protection Mode alarm.
Type: Alphanumeric value set.
Range: LOOPBACK, SHORTENED, TOPEX.
Default: LOOPBACK.

7. **rcamode** – Ring Circuit Alarm Mode
This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE

8. **swtr** – Span Wait To Restore Time
The parameter specifies the wait to restore time for span switches used by the protection state machine operating in revertive mode. The protection shall operate in revertive mode if the wait-to-restore time is in the range 0-12. The protection shall operate in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

9. **westwkg** – West Worker Port
This parameter specifies the port being a member in the role of the west worker port for a 4-fibre MS-SPRing/BLSR protection group
Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID
Range: In case of an asymmetric ring: NOTUSED; in all other cases: SPortAID.

10. **wtr** – Wait To Restore Time
This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
Note: If wtr time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

Spec block parameters (path protection group)

1. **`dripathpn`** – SNCP-UPSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type SNCP-UPSR.
   Type: Alphanumeric string.
   Range: 1-24 characters including the character "-".
   Default: DEFAULT.
2. **`htmr`** – Hold-Off Time
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms].

   Range for high order path protection groups, the range is:

   - 0 .. 99 ms in 1 ms steps and
   - 00 .. 10000 ms in100 ms steps

   Range for low order path protection groups, the range is:

   - 0, 25, 50, 75 ms (thus in steps of 25 ms) and
   - 100 .. 10000 ms in 100 ms steps.
3. **`newppgname`** **–** New Path protection group Name
   This parameter specifies the path protection group name. Storage and retrieval of the *ppgname* is case sensitive, but actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
   Type: text string.
   Range: 0 to 24 characters.
   Default: empty string
4. **`rme`** – Protection Reversion
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group.
   The value ENABLE is used if the protection operates in revertive mode.
   The value DISABLE is used if the protection operates in non-revertive mode.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
5. **`snctype`** **–** SNC Protection Type
   This parameter specifies the type of the path protection w.r.t. switch criteria used by the protection state machine.
   The value SNCI (Inherent) is used if server signal fail is used as switch criteria.
   The value SNCN (Non-Instrusive) is used if server signal fail and defects

supervised by a non-intrusive path monitor are used as switch criteria
The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
Type: Alphanumeric value set.
Range: SNCI, SNCN, NA.

6. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
   Type: numeric string [minute].
   Range: 0 – 60.

Note:

If a value beyond the allowed range is entered for the hold-off time (*htmr*) or for the wait-to restore time (*wtr*) the returned error code is incorrect: Instead of `/* Input, Data Not Valid */` the following error code is returned: `/*Input, Non-null Unimplemented Parameter*/`

Output format

After receiving the **`ED-PROTN-GRP`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response.

Detailed behavior description

The command modifies attributes for the existing protection group instances identified by *aid, rid, ppgname* of type as specified by means of parameter *protype*. The corresponding attribute of the protection group instance is updated with the value as specified in command block.

The command parameter *newrid* causes an update of the attribute *rid*. The command parameter *newppgname* causes an update of the attribute *ppgname*.

The update of the attributes is reported by means of an ED-PROTN-GRP notification.

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* to identify path protection groups is not recommended.

DCC protection as part of 1+1 multiplex section/line protection

If DCC protection is enabled for section/rs dcc channels (value of dccsrsprotmd equals ENABLE), then following updates for DCC channels are done.

1. The protection mode (attribute dccprottp) of the dcc channel of type 'SECTION' associated with the worker port is set to '1+1' protected. This update is reported by means of a ED-FECOM notification for the changed attribute.

If DCC protection is disabled for section/rs dcc channels (value of dccsrsprotmd equals DISABLE), then following updates for DCC channels are done.

1. The protection mode (attribute dccprottp) of the dcc channel of type 'SECTION' associated with the worker port is set to '0X1' unprotected. This update is reported by means of a ED-FECOM notification for the changed attribute.

If DCC protection is enabled for line/ms dcc channels (value of dcclmsprotmd equals ENABLE), then following updates for DCC channels are done.

1. The protection mode (attribute dccprottp) of the dcc channel of type 'LINE' associated with the worker port is set to '1+1' protected. This update is reported by means of a ED-FECOM notification for the changed attribute.

If DCC protection is disabled for line/ms dcc channels (value of dcclmsprotmd equals DISABLE), then following update for DCC channels are done.

1. The protection mode (attribute dccprottp) of the dcc channel of type 'LINE' associated with the worker port is set to '0X1' unprotected. This update is reported by means of a ED-FECOM notification for the changed attribute.

Path protection group

Identification of VCG tributaries: The system shall allow to edit path protection groups at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

Example input/output

Multiplex section/line protection group 1+1

The following example assigns a wait-to-restore time of 8 minutes to the protection group named 1-1-o011 which is of type 1+1:

```
ED-PROTN-GRP:LT-UNITE:1-1-o011:123456::1+1:wtr=8;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

2 fibre MS-SPRing/BLSR protection group

The following example assigns a wait-to-restore time of 10 minutes to the two fiber protection group that is identified by ring id *denver21*. It also renames the ring id to *dallas3*:

```
ED-PROTN-GRP:LT-UNITE::123456::2F,”denver21”:newrid=”dallas3”,wtr=10;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

4 fibre MS-SPRing/BLSR protection group

The following example assigns a wait-to-restore time of 10 minutes to the four fiber protection group that is identified by ring id *denver2*1. It also renames the ring id to *dallas3*:

```
ED-PROTN-GRP:LT-UNITE::123456::4F,”denver21”:newrid=”dallas3”,wtr=10;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Path protection group

The following example assigns a holdoff time of 100 mesc to the path protection group that is identified by path protection group name *denver21*. It also renames the path protection group name to *dallas3*:

```
ED-PROTN-GRP:LT-UNITE::123456::PATHDRI,, ”denver21”:newppgname=”dallas3”,htmr=10;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

The following example assigns a holdoff time of 100 mesc to both path protection groups forming an emulated selector bridge on input tributaries 1-1-#-#-8-1-25 and 1-1-#-#-6-1-25. It also renames the path protection group name to dallas3:

```
ED-PROTN-GRP:LT-UNITE:1-1-#-#-8-1-25:123456::PATHDRI,,,
1-1-#-#-6-1-25:newppgname=”dallas3”,htmr=10;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses (Multiplex section/Line protection group)

Validation rule: dcc protection side is enabled

The dcc channel (section/line) of the protection side is disabled if corresponding dcc protection is enabled (section/line).

Error code:

```
SNVS – Status, Not in Valid State, protection dcc channel is enabled
```

Validation rule: dcc protection: working side is disabled

The dcc channel (section/line) of the working side is enabled if corresponding dcc protection is enabled (section/line).

Error code:

`SNVS – Status, Not in Valid State, working dcc channel is disabled`

Validation rule: dcc protection: protection side is active

The system shall reject the disabling of the dcc protection, if the protection side of dcc protection is the active one.

Error code:

`SNVS – Status, Not in Valid State, dcc protection side carries traffic`

Validation rule: Transparent DCC for DCC Line/MS protection mode

The LINE DCC channel of protecting port is not used in a transparent DCC cross-connection if DCC Line/MS Protection Mode – *dcclmsprotmd* – will be set to ENABLED.

Error code:

`SNVS – Status, Not in Valid State, DCC channel used in transparent DCC cross-connection`

Validation rule: Transparent DCC for DCC Sect/RS protection mode

The SECTION DCC channel of protecting port is not used in a ransparent DCC cross-connection if DCC Sect/RS Protection Mode – dcslrsprotmd – will be set to ENABLED.

Error code:

`SNVS – Status, Not in Valid State, DCC channel used in transparent DCC cross-connection`

Error responses (2-fibre MS-SPRing/BLSR protection group)

Validation rule: unique rid

The specified *newrid* is not already used by an already existing protection group of type 2F, 4F.

Error code:

`IDNV – Input, Data Not Valid, rid is already used by another protection group`

Validation rule: consistent aid and rid

If both an *aid* and an *rid* are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: Automatic squelch map calculation only with automatic ring discovery

The system shall not allow to set *ard* to DISABLE while the Automatic Squelch Map Calculation (asmc) is enabled or to set *asmc* to ENABLE while the Automatic Ring Discovery (ard) is disabled.

Error code:

`IDNV – Input, Data Not Valid, Automatic Squelch Map Calculation only allowed with Automatic Ring Discovery`

Error responses (4-fibre MS-SPRing/BLSR protection group)

Validation rule: unique *rid*

The specified *newrid* is not already used by an already existing protection group of type 2F, 4F.

Error code:

`IDNV – Input, Data Not Valid, rid is already used by another protection group`

The check of *rid* uniqueness is case insensitive.

Validation rule: port on same pack

Each port being added as member for the protection group is located on different circuit packs.

Error code:

`SNVS – Status, Not in Valid State, ports must be on different circuit packs`

Validation rule: common interface standard

Each port being added as member for the protection group shall have the same interface standard as already included ports.

Error code:

`SNVS – Status, Not in Valid State, ports have different interface types`

Validation rule: port in one group

Each port being added as member for the protection group is not be a member of an already provisioned protection group.

Error code:

`SNVS – Status, Not in Valid State, port is already used in another protection group`

Validation rule: common port rate

Each port being added as member for the protection group shall have the same port rate as the other ports.

Error code:

`SNVS – Status, Not in Valid State, ports have different port rates`

Validation rule: no XC loopback on tribs

There shall exist no cross-connection loopbacks on tributaries related to the specified ports being added/removed as member for the protection group.

Error code:

`SNVS – Status, Not in Valid State, Tributary is used in cross-connection`

Validation rule: No HO/LO cross-connection or LO substructure at tributaries

There shall exist no HO or LO cross-connection or LO substructure on tributaries related to the specified ports being added/removed as member for the protection group.

Error code:

`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

Validation rule: no test access session active

Tributaries related to the specified ports being added/removed as member for the protection group must not be used in a test access session either as tributaries under test or as test access tributaries.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: dcc enabled on worker port

If automatic ring discovery (parameter ard) is ENABLED, then for each working port being added as member for the protection group at least one dcc channel (section/dcc-r or line/dcc-m) with mode UITS is enabled.

Error code:

`SNVS – Status, Not in Valid State, no dcc channel with mode UITS enabled`

Validation rule: dcc enabled on protection port

If the worker port is removed from one side of a protection group while the parameter ard (automatic ring discovery) is ENABLED, then at least one dcc channel (section/dcc-r or line/dcc-m) for the protection port at the same side is enabled.

Error code: `SNVS – Status, Not in Valid State, no dcc channel enabled`

Validation rule: Automatic squelch map calculation only with automatic ring discovery

The system shall not allow to set *ard* to DISABLE while the Automatic Squelch Map Calculation (asmc) is enabled or to set *asmc* to ENABLE while the Automatic Ring Discovery (ard) is disabled.

Error code:

`IDNV – Input, Data Not Valid, Automatic Squelch Map Calculation only allowed with Automatic Ring Discovery`

Validation rule: Working port to be added is assigned as timing reference

If a port that is already assigned as timing reference is added to a protection group as working port, then no port of the circuit pack that holds the protection port is assigned as timing reference.

Error code:

`SNVS – Status, Not in Valid State, one port of protection unit is assigned as timing reference`

Validation rule: Timing Reference Assigned on Protection Unit

If a port which is not assigned as timing reference is added to a protection group as working port, then the protection port is not assigned as timing reference.

Error code:

`SNVS – Status, Not in Valid State, protection port is assigned as timing reference`

Validation rule: consistent *aid* and *rid*

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: no multiple added XCs for TOP

If *protnmode* is changed to SHORTENED or TOPEX and there exists a path protection group having one trib of worker port as output (so-called multiple added XC) then the following error response is returned.

Error code:

`SNVS – Status, Not in Valid State, multiple added XCs for MS-SPRing TOP`

Validation rule: SDH ports for TOP

If *protnmode* is changed to SHORTENED or TOPEX then all member ports shall have the interface standard SDH.

Error code:

`SNVS – Status, Not in Valid State, SONET port not allowed for TOP`

Validation rule: No cross-connections allowed when leaving TOPEX

The system shall reject all ED-PROTN-GRP commands for changing the parameter protnmode to any value beside TOPEX if the current value of the protection group is TOPEX and the protection ports of the protection group are involved in any cross-connection.

Error code:
`SNVS – Status, Not in Valid State, extra traffic cross-connections are still existing`

Validation rule: Idle ring node for protocol conversion

If protnmode is changed then ringndst is ’IDLE’ or ’SUSPENDED-IDLE’.

Error code: `SNVS – Status, Not in Valid State, node not in idle state`

Validation rule: supported *portClass* for TOP

If *protnmode* equals SHORTENED or TOPEX (both are TOP), then none of the specified member ports shall have the *portClass* EDGE.

Error code:
`SNVS – Status, Not in Valid State, portClass must not be EDGE for TOP`

Validation rule: portClass not INNI

Ports must not be of portClass INNI.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Unallowed modification of worker ports

The system shall only allow to change the parameters eastwkg and westwkg from a port AID to NOTUSED or from NOTUSED to a port AID (i.e. asymmetric ring conversion). All other modification attempts, i.e. change from one port AID to another port AID is denied.

Error code:
`IDNV – Input, Data Not Valid, change of worker port not allowed`

Error responses (path protection group)

Validation rule: Protection Group must exist

If the parameter *aid*, or the combination of the parameters *aid* and *ppgname* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Protection Group must exist

If the combination of the parameters *aid* and *aid1* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Maximum number of ppgnames exceeded

The new *ppgname* shall not exceed the maximum number of 768 ppgnames (i.e. is already in use for 768 other path protection groups).

Error code:

`SSRE – Status, System Resources Exceeded, limit of ppgnames exceeded`

Validation rule: *ppgname* and low order path protections

If a *ppgname* is given as input parameter while the *aid* references a low order tributary, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, ppgname not supported for this transmission rate`

Validation rule: SNC Type change on higher order path protection groups sharing input tributaries

The system shall allow to change the SNC type of a higher order path protection group only if one of the following conditions is fulfilled:

- None of the two input tributaries of the path protection group is used as input tributary of another path protection group.
- Both input tributaries of the path protection group are used as input tributaries of the same colocated path protection group.

Error code:

`SNVS – Status, Not in Valid State, Input tributary shared with other path protection group`

Validation rule: Access via tributary on INNI port

If in input parameter *aid* or *aid1* for access via input or output a tributary is given located on a port of port class INNI, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, No Access via AID located on INNI port`

Related TL1 messages

- "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
- "DLT-PROTN-GRP: Delete protection group" (p. 3-233)
- "OPR-PROTNSW: Operate protection switch" (p. 3-2057)
- "RLS-PROTNSW: Release protection switch" (p. 3-2203)

ED-PROTN-GRP notification

Output format

The following syntax is used for 1+1 multiplex section/line, 2-/4-fibre MS-SPRing/BLSR, path protection group.

General Syntax: **`ED-PROTN-GRP`**:*aid*:*protype:spec_block*;

Path Protection Group Syntax: **`ED-PROTN-GRP`**:*aid:protype:spec_block {outtribaid, CML: cml_block}+* ;

Output parameters

Command parameters (multiplex section/line protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port aid being a member of the group.
   Type: AID
   Range: *bay-shelf*-oall (ranged over OnePlusOne_ProtGrpAID) or OnePlusOne_ProtGrpAID (not ranged) or NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged)
2. **`protype`** – Protection Type
   The parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1x***N***OPT (only R3.0 and later).
   In mind the ***N*** in “1x***N***OPT” is considered to be always equal “1”.
   Type: Alphanumeric value set.
   Range: 1+1 (default) or 1xNOPT.
   Default: 1+1

Command parameters (2-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID
   Range: *bay-shelf*-tall (ranged over TwoFibre_ProtGrpAID) or bay-shelf-fall (ranged over FourFibre_ProtGrpAID) or TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged).
2. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F

Command parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID
   Range: *bay-shelf*-tall (ranged over TwoFibre_ProtGrpAID) or bay-shelf-fall (ranged over FourFibre_ProtGrpAID) or TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged).
2. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F

Command parameters (path protection group)

1. **`aid`** – Access Identifier
   The parameter specifies the AID of the path protection group.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.
2. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: PATHDRI
3. **`outtribaid`** – Output Tributary Access Identifier
   This parameter specifies the output tributary AID for a constituent signal within the protection group. One value is returned for each constituent signal together with the corresponding CML (Constituent Member List) block.
   Type:AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.

Spec block parameters (multiplex section/line protection group)

1. **apspn** – MSP-APS ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MSP-APS (Multiplex Section Protection – Automatic protection switching).
   Type: Alphanumeric string
   Range: 1-24 characters including the character "-".
   Default: DEFAULT
2. **dcclmsprotmd** – DCC Line/MS Protection Mode
   The parameter specifies, whether the line/MS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set 3;
   Range: ENABLE, DISABLE
   Default: DISABLE
3. **dccsrsprotmd** – DCC Sect/RS Protection Mode
   The parameter specifies, whether the section/RS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set
   Range: ENABLE, DISABLE
   Default: DISABLE
4. **wtr** – Wait To Restore Time / Wait to Rename Time
   The parameter specifies the "wait to restore" time used by the protection state machine operating in revertive mode. In case of MSP 1+1 optimized, the same parameter is used for the "wait to rename" time.
   Type: numeric string [minute]
   Range: 0 – 60
   Default: 5.

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

1. **ard** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value manually with ED-MAP-RING and supply the ring map information to each node on the ring for the BLSR protocol to operate. It is necessary to activate this user-provisioned information with

SET-MAP-RING. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.

2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: Alphanumeric string
Range: ENABLE, DISABLE (default)
3. **`blsrpn`** – MS-SPRing-BLSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
Type: Alphanumeric string
Range: 1-24 characters including the character "-".
Default: DEFAULT
4. **`newrid`** – New Ring Identification
This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE.
5. **`rcamode`** – Ring Circuit Alarm Mode
This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. I f *rcamode* is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE
6. **`wtr`** – Wait To Restore Time
This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.

Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value and supply the ring map information to each node on the ring for the BLSR protocol to operate. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.
2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: Alphanumeric string
Range: ENABLE, DISABLE (default)
3. **`blsrpn`** – MS-SPRing-BLSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
Type: Alphanumeric string.
Range: 1-24 characters including the character “-”.
Default: DEFAULT.
4. **`newrid`** – New Ring Identification
This parameter specifies the new ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the *rid* to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE
5. **`protnmode`** – Protection Switch Mode
The parameter specifies the protection switch mode. The value LOOPBACK is used for the classical ring application. The value SHORTENED is used for transmission delay sensitive applications. The value TOPEX (only R3.0 - R3.2 and later) is used for the so called transoceanic application, which allows extra traffic, improves restorability and improves delay as well as in SHORTENED mode.
In case of a ring switch … LOOPBACK performs the bridge and switch function in

that nodes which are most close to the failed span. SHORTENED performs the bridge function at the add node repectively the switch function at the drop node of each ring circuit. It is an application only similar to the transoceanic protocol. TOPEX allows, compared to SHORTENED mode, additionally the traffic restoration of not affected spans. It is the transoceanic protocol application. Consistent Protection Switch Mode Setting: The Protection Switch Mode must be set consistently on all nodes in the ring. E.g. for all UNITE NEs in a TOP+EX ring the parameter protnmode must be set to TOPEX. In a ring with inconsistent Protection Switch Mode, there will be no protection switching. The UNITE NEs will raise an Inconsistent Ring Protection Mode alarm.
Type: Alphanumeric value set.
Range: LOOPBACK, SHORTENED, TOPEX.
Default: LOOPBACK.

6. **`rcamode`** – Ring Circuit Alarm Mode
   This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
7. **`swtr`** – Span Wait To Restore Time
   The parameter specifies the wait to restore time for span switches used by the protection state machine operating in revertive mode. The protection shall operate in revertive mode if the wait-to-restore time is in the range 0-12. The protection shall operate in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
   Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
   Type: numeric string [minute].
   Range: 0 – 12 in 1 minute increments and 99.
   Default: 5.
8. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
   Note: If wtr time is set to infinite, the switch request state is WTR until a higher request occurs.
   Type: numeric string [minute].
   Range: 0 – 12 in 1 minute increments and 99.
   Default: 5.

Spec block parameters (Path protection group)

1. **dripathpn** – SNCP-UPSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type SNCP-UPSR
   Type: Alphanumeric string
   Range: 1-24 characters including the character "-".
   Default: DEFAULT
2. **htmr** – Hold-Off Time
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms].
   Range: 0 .. 99 ms in 1 ms steps and 100 .. 10000 ms in 100 ms steps.
3. **newppgname** – New Path protection group Name
   This parameter specifies the path protection group name. Storage and retrieval of the *ppgname* is case sensitive, but actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
   Type: text string.
   Range: 0 to 24 characters.
   Default: empty string
4. **protn** – Protection Tributary
   This parameter specifies the protection input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID, VCGNumAID, GenLOTribAID, GenLOTribAID, TMuxAPTNumAID.
   Default: None.
5. **rme** – Protection Reversion
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group.
   The value ENABLE is used if the protection operates in revertive mode.
   The value DISABLE is used if the protection operates in non-revertive mode.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
6. **snctype** – SNC Protection Type
   This parameter specifies the type of the path protection w.r.t. switch criteria used by the protection state machine.
   The value SNCI (Inherent) is used if server signal fail is used as switch criteria.
   The value SNCN (Non-Instrusive) is used if server signal fail and defects supervised by a non-intrusive path monitor are used as switch criteria
   The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
   Type: Alphanumeric value set.
   Range: SNCI, SNCN, NA.

7. **`wkg`** – Worker Tributary
   This parameter specifies the worker input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID, VCGNumAID, GenLOTribAID, GenLOTribAID, TMuxAPTNumAID.
   Default: None.
8. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
   Type: numeric string [minute].
   Range: 0 – 60.

CML block parameters (Path protection group)

1. **`cst`** – Constituent Signal Type
   The parameter specifies the rate of the constituent signal within the path protection group.
   Type: Alphanumeric string
   Range: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C
   Default: The default value is derived from the tributary input signal rate list in case of fixed mode.

Example output

Multiplex section/line protection group (1+1)

The following example reports the modification of wait-to restore timer for an 1+1 multiplex section/line protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ”TIME=15-00-00,DATE=00-11-03:ED-PROTN-GRP:1-1-o001:1+1:wtr=8”
;
```

2 fibre MS-SPRing/BLSR protection group

The following example reports the modification of wait-to-restore time and ring id for a two fiber protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ”TIME=15-00-00,DATE=00-11-03:ED-PROTN-GRP:1-1-t001:2F:newrid=\”dallas3\”,wtr=10”
;
```

4 fibre MS-SPRing/BLSR protection group

The following example reports the modification of wait-to-restore time and ring id for a four fiber protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ED-PROTN-GRP:1-1-f001:4F:newrid=\″dallas3\″,wtr=10″
;
```

Path protection group

The following example reports the modification of hold off time and path protection group name for a path protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ED-PROTN-GRP:1-1-#-#-10-1-
      25:PATHDRI:newppgname=\″dallas3\″,htmr=10″
;
```

ED-PTF: Edit tributary with Path Termination Function (PTF)

The **ED-PTF** command modifies the properties of a tributary with path termination function.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-PTF**-*modifier:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range:
   - STS for DS3 card
   - HOSTS1 for LOXC card
   - HOVC4 for LOXC card
   - TMASTS1 for TransMUX asynchronous port
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   The access identifier specifies the PTF tributary whose attributes are reported.
   Type: AID
   Range:
   - PTNumAID for modifier: STS, HOSTS1, HOVC4 as defined in AID structure. That means the value **all** is allowed for Simple_PTNumAID.
   - TMuxAPTNumAID for modifier: TMASTS1

   For Simple_UnivSlotAID there is no **usall** value. For Simple_TMuxAPortAID there is no **all** value.
4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   The parameter *brstintvl* specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST, then *brstintvl* has no affect to NE behavior.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *brstintvl* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Option: If the parameter *brstintvl* is not supported, then the parameter is omitted.
   Type: Integer.
   Range: 2, 3, … 10 [seconds]
   Default: 7 [seconds].

2. **`brstthr`** – Bursty Error Threshold
   The parameter *brstthr* specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *brstthr* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Option: If the parameter *brstthr* is not supported, then the parameter is omitted.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30.

3. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to

RTRV-PTF Output parameters.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter *degthr* is not supported for the following modifiers:

- TMSSTS1

Type: Integer.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

4. **`dexcthr`** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of sdsfmode refer to RTRV-PTF Output parameters.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter *dexcthr* is not supported for the following modifiers:

- TMSSTS1

Type: Integer.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

5. **`exphptrc`** – Expected HP trace identifier
This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if ptrcrfmt is set to 16.

Limitation: The parameter is not supported for modifiers:

- TMSSTS1

Type: Hexadecimal number without a preceding “H”.
Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
Default: 89000000000000000000000000000000

6. **`feppm`** – Far End Path PM Enable
The parameter *feppm* specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the parameters for far-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1

Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

7. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion

   The parameter *hptimais* shall specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

   The parameter is not supported for the following modifiers:

   - TMSSTS1

   Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE.

   Default: The default value depends on interface standard specified by *ptfstd* for the PTF tributary. DISABLE, if *ptfstd* = SONET

8. **`hptimmon`** – HP Trace Identifier Mismatch Monitor

   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

   The parameter is not supported for the following modifiers:

   - TMSSTS1

   Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE (default).

9. **`inputsig`** – Tributary Input Signal Rate List

   This parameter specifies the input LO signal structure.

   The parameter *inputsig* is only supported for the following modifiers:

   - HOVC4
   - HOSTS1
   - TMSSTS1

   Option: If the parameter *inputsig* is not supported, then the parameter is omitted.
   Type: numeric value set
   Range: NX-NX-… (N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
   The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33

The possible values (SONET) are: 2811
Restricted Range with AU3/AU4 Conversion cross-connections:
If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *inputsig* is restricted to: 33.
Restricted Range with TransMUX:
If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.

Default:

- 6312 for SDH
- 2811 for SONET

10. **neppm** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the possibility to enable near-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1

Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE.

Default: DISABLE.

11. **pmhpfeses** – PM HP Far End SES Threshold
This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *ptfstd* for the PTF tributary. If *ptfstd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the possibility to enable far-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1

Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

12. **`pmhpneses`** – PM HP Near End SES Threshold
    This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *ptfstd* for the tributary. If *ptfstd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
    This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
    Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1

    Type: Integer.
    Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
    Default: 2400

13. **`ptca`** – Path TCA Profile Pointer
    This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    TCA profile: The TCA profile type depends on the type and standard of the PTF, which is specified with the parameter *modifier* or *aidtype*:

    - PATH if *modifier* = TMASTS1

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1
    - HOVC4

    Type: Alphanumeric value set.
    Range: 1 … 24 characters.
    Default: DEFAULT0

14. **`ptfstd`** – Path Termination Standard
    Standard of the tributary with Path Termination Function.
    This is currently a read-only parameter.
    Type: Alphanumeric value set.

    Range:

    - SDH (only if *modifier* = HOVC4)
    - SONET (only if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1)

    Default:

    - SDH (if if *modifier* = HOVC4)
    - SONET (if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1).

15. **`ptpn`** – Path Terminating ASA Profile Name
    This parameter points to a tributary ASAP profile name *pfname* for the designated AID.

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
    Range: 1 … 24 characters
    Default: DEFAULT

16. **`ptrcrfmt`** – Path Trace Read Format
    The parameter *ptrcrfmt* specifies the format of the path trace (J1 byte) to be received (sink).

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Type: Numeric string.

    Range:

    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

17. **`ptrcwfmt`** – Path Trace Write Format
    The parameter *ptrcwfmt* specifies the format of the path trace (J1 byte) to be sent (source).

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Type: Numeric string.

    Range:

    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

    Default: 2.

18. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    This parameter specifies the determination mode of degraded defect for transmission.
    Limitation for PTF related to modifiers: Near-end performance monitoring is not available for modifier STS. The parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1

- HOVC4
- HOSTS1

Option: If the parameter *sdsfmode* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: BURST, POISSON
Default: The default value depends on interface standard specified by *ptfstd* for the tributary. POISSON, if *ptfstd* = SONET.

19. **`tmonmd`** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Alphanumeric value set.
    Range: MON, NMON.
    Default: NMON.
20. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Hexadecimal number without a preceding “H”.
    Range: the range depends on the parameter ptrcwfmt specified in RTRV-PTF Output parameters
    - length: 32 hexadecimal digits, if *ptrcwfmt*=16; 128 hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt*=16; 64 byte string if *ptrcwfmt* = 64

    Default: 00. This implies also that *ptrcwfmt* is set to 2.
21. **`tslmmon`** – Trail Signal Label Mismatch Monitor
    The parameter *tslmmon* specifies the mode of the TSL mismatch detection.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Alphanumeric value set.
    Range: DISABLE, ENABLE.
    Default: ENABLE.
22. **`unequiposig`** – Unequipped Output Signal List
    The parameter *unequiposig* specifies the output LO signal structure, independent whether the signal is connected or not. The parameter *unequiposig* is only supported for modifiers HOVC4. HOSTS1. and TMSSTS1.
    Option: If the parameter *unequiposig* is not supported, then the parameter is

omitted.
Type: numeric value set
Range: NX-NX-…(N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
The possible values (SONET) are: 2811
Restricted Range with AU3/AU4 Conversion cross-connections:
If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *unequiposig* is restricted to: 33.
Restricted Range with TransMUX:
If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.
Default:

- 6312 for SDH
- 2811 for SONET

Output format

If the network element fully complies with the **ED-PTF** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The ED-PTF shall generate a REPT DBCHG message. The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed entity according to the parameters given with this command. The parameters *brstintvl*, *brstthr*, *degthr* and *dexcthr* is modifiable also if they are not meaningful at the moment. The modification is reported by means of an ED-PTF notification.

Dependencies

- *brstintvl* and *brstthr* are only meaningful if *sdsfmode* is set to BURST.
- *degthr* and *dexcthr* are only meaningful if *sdsfmode* is set to POISSON.

Supply outgoing J1 trace value and format together

When changing the format of the outgoing J1 trace *ptrcwfmt*, the parameter *trc* specifying the outgoing J1 trace shall always be supplied within the same TL1-command to always have a consistent set of trace format and value. When the format of the outgoing J1 trace *ptrcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J1 trace (mandatory parameter trc) specified in the command is ignored.

Supply outgoing J1 trace value according to format of outgoing J1

When changing the outgoing J1 trace value, the format must be according to the format parameter ptrcwfmt for the outgoing J1 trace.

Change of J1 expected format - FM impact

When the expected J1 format (ptrcrfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J1 (exphptrc) and the currently received J1 (ptrc).

When the expected J1 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of tributary standard - Notification behavior

A change of the interface standard ptfstd is not allowed unless this parameter is set to the same value. For convenience reasons, the parameter ptfstd is not dropped from the message definition as other products managed by the same EMS may support this common parameter.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Changing LO substructure (SDH only)

The system shall update the parameters *inputsig* and *unequiposig*. Changes of the inputsig and unequiposig values are notified via ED-PTF notifications. If one or both of the parameters *inputsig* and *unequiposig* are not specified in the command or are set to the same value as already provisioned, then the system shall not change the parameter value(s).

Update affected entities

- If one third of the substructured VC4 tributary is changed from 21 LO VC12 to 1 LO VC3 then the parameters of the first LONIM entity of that third is set to default values according to the newly represented transmission rate. The parameters neppm, feppm, tcamd and ptca of the first LONIM entity of that third is not changed.. The parameters of the other 20 LO tributaries shall remain unchanged.
- If one third of the substructured VC4 tributary is changed from 1 LO VC3 to 21 LO VC12 then the parameters of all 21 LONIM entities of that third is set to default values according to the newly represented transmission rate. The parameters neppm, feppm, tcamd and ptca of the 21 LONIM entities of that third shall remain unchanged.

Changes of LO tributary parameter values are notified via ED-LONIM Notifications. Group notifications are used if a contiguous range of LO tributaries is affected in the same way (i.e. AID ranging is applicable).

Example input/output

```
ED-PTF-STS:LT-UNITE:1-1-#-#-21-1-1:123456:::tmonmd=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-PTF`** command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: supply of transmitted path trace

Check that a transmit path trace (trc) is supplied, too, if the path trace mode (ptrcwfmt) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trc must also be supplied for a path trace mode change`

Validation rule: transmitted path trace length

Check that a length of the transmit path trace (trc) fits to the length given by path trace mode, i.e. write format (ptrcwfmt).

Error code:

`IDNC – Input, Data Not Consistent, trc length must match the given path trace mode`

Validation rule: Substructure change not allowed while conversion cross-connections exist

The command processing shall not allow to modify the substructure (parameters *inputsig* and *unequiposig*) when AU3/AU4 conversion cross-connections exist at LO tributaries related to the PTF.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: Substructure change not allowed with LO cross-connections at input tributaries

If inpusig is set to a new value, while LO cross-connections exist at input tributaries of the affected third of the substructure, then the command is denied

Error code:

`SNVS – Status, Not in Valid State, lower order cross-connections are still existing`

Validation rule: Substructure change not allowed with LO cross-connections at output tributaries

If *unequiposig* is set to a new value, while LO cross-connections exist at output tributaries of the affected third of the substructure, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, lower order cross-connections are still existing`

Validation rule: Maximum number of enabled PM monitors

For PTF entities with the modifier STS, check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

This validation rule is not applicable for entities related to lower order tributaries (PTF and LONIM) as all possible monitors on these entities are allowed to be activated in parallel.

This validation rule is not applicable for entities related to TransMUX tributaries (PTF) as all possible monitors on these entities are allowed to be activated in parallel.

Validation rule: Check CCB against *ptfstd* of related tributary

Check wether the CCB fits to *ptfstd* of the tributary.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

Example: DENY, if a command with modifier = VC is made on a port which is provisioned as *ptfstd* = SONET.

Related TL1 messages

- "RTRV-PTF: Retrieve tributary with path termination function" (p. 3-2800)

ED-PTF notification

Output format

Syntax

ED-PTF:*aid,aidtype::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   The access identifier specifies the PTF tributary whose attributes are reported.
   Type: AID
   Range:
   - PTNum for modifier: STS, HOSTS1, HOVC4, ALL
   - TMuxAPTNumAID for modifier: TMASTS1, ALL
   - TMuxSPTNumAID for modifier: TMSSTS1, ALL

   as defined in AID structure with exception of value *all*.
2. **aidtype** – Access Identifier Type
   The *aidtype* specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range:
   - STS
   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Default: None.

Spec block parameters

1. **brstintvl**– Bursty Interval
   The parameter *brstintvl* specifies the measurement period for degraded defect if *sdsfmode* is set to BURST.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.
   The parameter *brstintvl* is not supported for the following modifiers:
   - HOSTS1
   - HOVC4

- TMASTS1
- TMSSTS1

Option: If the parameter *brstintvl* is not supported, then the parameter is omitted.
Type: Integer.
Range: 2, 3, … 10 [seconds]
Default: 7 [seconds].

2. **`brstthr`** – Bursty Error Threshold
   The parameter *brstthr* specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *brstthr* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Option: If the parameter *brstthr* is not supported, then the parameter is omitted.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30.

3. **`crslpbkstat`** – Cross-connect Loopback Status
   The parameter *crslpbkstat* reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).

   The parameter *crslpbkstat* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMSSTS1

   Option: If the parameter *crslpbkstat* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: YES, for enabled loopback; NO, for disabled loopback

4. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to RTRV-PTF Output parameters.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter *degthr* is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Integer.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

5. **`dexcthr`** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of sdsfmode refer to RTRV-PTF Output parameters.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.
The parameter *dexcthr* is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
Type: Integer.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

6. **`exphptrc` –** Expected HP trace identifier
This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
Limitation: The parameter is not supported for modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
Type: Hexadecimal number without a preceding “H”.
Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
Default: 89000000000000000000000000000000

7. **`feppm` –** Far End Path PM Enable
The parameter *feppm* specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the parameters for far-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

8. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion
The parameter *hptimais* shall specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE.

Default: The default value depends on interface standard specified by *ptfstd* for the PTF tributary. DISABLE, if *ptfstd* = SONET

9. **`hptimmon`** – HP Trace Identifier Mismatch Monitor
This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **`inputsig`** – Tributary Input Signal Rate List
This parameter specifies the input LO signal structure.

The parameter *inputsig* is only supported for the following modifiers:

- HOVC4
- HOSTS1
- TMSSTS1

Option: If the parameter *inputsig* is not supported, then the parameter is omitted.
Type: numeric value set
Range: NX-NX-… (N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
The possible values (SONET) are: 2811
Restricted Range with AU3/AU4 Conversion cross-connections:
If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *inputsig* is restricted to: 33.
Restricted Range with TransMUX:
If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.

Default:

- 6312 for SDH
- 2811 for SONET

11. **neppm** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the possibility to enable near-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE.

Default: DISABLE.

12. **pmhpfeses** – PM HP Far End SES Threshold
This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *ptfstd* for the PTF tributary. If *ptfstd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.

This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the possibility to enable far-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

13. **`pmhpneses`** – PM HP Near End SES Threshold
    This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *ptfstd* for the tributary. If *ptfstd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
    This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
    Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Integer.
    Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
    Default: 2400

14. **`ptca`** – Path TCA Profile Pointer
    This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    TCA profile: The TCA profile type depends on the type and standard of the PTF, which is specified with the parameter *modifier* or *aidtype*:

    - PATH if *modifier* = TMASTS1

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1
- HOVC4

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.
Range: 1 … 24 characters.
Default: DEFAULT0

15. **`ptfstd`** – Path Termination Standard
Standard of the tributary with Path Termination Function.
This is currently a read-only parameter.
Type: Alphanumeric value set.
Range:
    - SDH (only if *modifier* = HOVC4)
    - SONET (only if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1)

    Default:
    - SDH (if if *modifier* = HOVC4)
    - SONET (if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1).
16. **`ptpn`** – Path Terminating ASA Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
Range: 1 … 24 characters
Default: DEFAULT
17. **`ptrcrfmt`** – Path Trace Read Format
The parameter *ptrcrfmt* specifies the format of the path trace (J1 byte) to be received (sink).
The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
Type: Numeric string.
Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

18. **`ptrcwfmt`** – Path Trace Write Format
The parameter *ptrcwfmt* specifies the format of the path trace (J1 byte) to be sent (source).

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16 byte specific string
- 64: for 64 byte string.

Default: 2.

19. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
This parameter specifies the determination mode of degraded defect for transmission.
Limitation for PTF related to modifiers: Near-end performance monitoring is not available for modifier STS. The parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1
- HOVC4
- HOSTS1

Option: If the parameter *sdsfmode* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: BURST, POISSON
Default: The default value depends on interface standard specified by *ptfstd* for the tributary. POISSON, if *ptfstd* = SONET.

20. **`substr`** – Substructured
The parameter *substr* indicates if the PTF is substructured.

Additional description for LOXC

This parameter indicates if the PTF is configured for AU3/AU4 conversion cross-connections.
The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created. A PTF, which is configured for AU3/AU4 conversion cannot be used to set up HO cross-connections.
The parameter *substr* is only supported, if *modifier* = STS.
If the parameter *substr* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- CNV: In use for HO side of a conversion cross-connection
- NO: Not used for conversion cross-connections

Default: NO.

21. **`tmonmd`** – Tributary Monitoring Mode
This parameter specifies the monitoring mode for the tributary.
The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range: MON, NMON.
    Default: NMON.

22. **`trc`** – Outgoing Path Trace
This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Hexadecimal number without a preceding "H".
    Range: the range depends on the parameter *ptrcwfmt* specified in RTRV-PTF Output parameters
    - length: 32 hexadecimal digits, if *ptrcwfmt* = 16; 128 hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt*=16; 64 byte string if *ptrcwfmt* = 64

    Default: 00. This implies also that *ptrcwfmt* is set to 2.

23. **`tslmmon`** – Trail Signal Label Mismatch Monitor
The parameter *tslmmon* specifies the mode of the TSL mismatch detection.
The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range: DISABLE, ENABLE.
    Default: ENABLE.

24. **`unequiposig`** – Unequipped Output Signal List
The parameter *unequiposig* specifies the output LO signal structure, independent whether the signal is connected or not. The parameter *unequiposig* is only supported for modifiers HOVC4. HOSTS1. and TMSSTS1.
Option: If the parameter *unequiposig* is not supported, then the parameter is

omitted.
Type: numeric value set
Range: NX-NX-…(N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
The possible values (SONET) are: 2811
Restricted Range with AU3/AU4 Conversion cross-connections:
If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *unequiposig* is restricted to: 33.
Restricted Range with TransMUX:
If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.
Default:

- 6312 for SDH
- 2811 for SONET

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-PTF:1-1-#-#-21-1-1,STS::tmonmd=MON″
;
```

ED-RC-GENERAL: Edit general routing controller settings

The **RTRV-RC-GENERAL** command is used to edit the general settings of the network element routing controller.

> **Important!** This functionality requires the software version R10.5.1. With older software versions it is not possible to configure parameters “Hello Interval” and “Dead Interval” to values different from their default values of 10 s (for “Hello Interval”) and 30 s (for “Dead Interval”).
>
> As corresponding parameters must be set to equal values for all RCs in the network, the whole network must be run with the default values. In case of a multi-vendor network, set the corresponding parameters of the non-1675 LambdaUnite MSS products to the default values of 1675 LambdaUnite MSS.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-RC-GENERAL**:*tid*::*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*.

Spec block parameters

1. **`hellointerval`** – Hello Interval
   This parameter is used to indicate the time (in seconds) between two succeeding "Hello" messages to be sent to a neighbor RC.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 10.
2. **`deadintv`** – Dead Interval
   This parameter is used to indicate the time (in seconds) that the system waits for the reception of "Hello" PDU from a neighbor RC before the neighbor RC is considered to be dead.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 30.

Output format

If the **`ED-RC-GENERAL`** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`ED-RC-GENERAL`** specifies the hello interval and the router dead interval. The hello interval is the time between two "Hello" PDU messages that are sent to an RC neighbor. The dead interval is the amount of time which must be passed without receiving a "Hello" PDU from an RC neighbor, before this RC neighbor is declared dead.

Example input/output

The following example shows the result of **`ED-RC-GENERAL`**:

```
ED-RC-GENERAL:NE-UNITE::123456:::hellointv=20, deadintv=60;
   LT-UNITE 00-12-24 12:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`RTRV-RC-GENERAL`** command.

Validation rule: RC must be created

If the RC is not created (through **`ACT-L1ROUTING`**), then the command is denied.

Error code: `IDNV – Input, Data Not Valid, RC not created`

Related TL1 messages

- “RTRV-RC-GENERAL: Retrieve general routing controller settings” (p. 3-2814)

ED-RC-GENERAL notification

Syntax

ED-RC-GENERAL:::*spec_block*

Output parameters

Spec block parameters

1. **hellointerval** – Hello Interval
   This parameter is used to indicate the time (in seconds) between two succeeding "Hello" messages to be sent to a neighbor RC.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 10.
2. **deadintv** – Dead Interval
   This parameter is used to indicate the time (in seconds) that the system waits for the reception of "Hello" PDU from a neighbor RC before the neighbor RC is considered to be dead.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 30.

Output format

If the **ED-RC-GENERAL** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Example input/output

```
   NE-UNITE 06-10-09 08:59:00
A  000 REPT DBCHG
   "TIME=08-59-00,DATE=06-10-09:ED-RC-GENERAL::: hellointv=20, deadintv=60";
```

ED-RSVP: Modify RSVP parameters

The **ED-RSVP** command modifies RSVP related parameters applicable for UNI or ENNI signaling on system level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-RSVP**-*modifier*:*tid*::*ctag*:::*spec_block;*

Input parameters

Command parameters

1. **modifier**
   The parameter modifier indicates the communication function entity for which the RSVP parameters apply.
   Type: Alphanumeric value set.
   Range: UNI or ENNI.
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*.

Spec block parameters

1. **hellointerval** – Hello Interval
   This parameter is used to indicate the time between two succeeding HELLO messages to be sent to any neighbor in hello interworkings.
   Type: Numeric string.
   Range: 1 … 60, indicating the Hello Interval in seconds
2. **maxhelloretransmit** – Maximum Hello Retransmissions
   This parameter is used to indicate the number of Hello Intervals without any HELLO_REQUEST or HELLO_ACK object received from a neighbor after which communication to this neighbor is considered as lost.
   Type: Numeric string.
   Range: 1 to 10.
   Default: 3
3. **srefreshtimer** – Srefresh Timer
   This parameter is used to indicate the maximum time between two consecutive SREFRESH requests for a UNI or network connection towards affected neighbors in srefresh interworkings.
   Type: Numeric string.
   Range: 10 … 180 indicating the Srefresh time in seconds.
   Default: 15.

Output format

If the **RTRV-RSVP** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command shall update the RSVP parameters for the node specified by the command modifier.

Example input/output

The following example shows the result of an **ED-RSVP** command:

```
ED-RSVP-UNI:LT-UNITE::123456:::maxhelloretransmit=8;

   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
```

```
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`RTRV-NE`** command.

Validation rule: Check of the maintenance condition

The system must be in maintenance condition (MCOND = YES). Otherwise the command is rejected.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: UNI node IP Address must be provisioned

If the command is executed with *modifier* = UNI while no valid UNI Node IP Address (uni_ip_ad) is provisioned, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, UNI Node IP address not provisioned`

Validation rule: ENNI node must be created

If the command is executed with *modifier* = ENNI while the ENNI Node is not created (by provisioning of the *sigPCIP_Address* in ENT-ULS), then the command is denied.

Error code: `SNVS – Status, Not in Valid State, ENNI Node not created`

Related TL1 messages

- “RTRV-RSVP: Retrieve RSVP parameters” (p. 3-2823)

ED-RSVP notification

Output format

Command syntax: **ED-RSVP**-*modifier*:::*spec_block;*

The output parameters *sid*, *date*, *time*, and atag are described in the REPT DBCHG Report Database Change notification.

Output parameters

Command parameters

1. **modifier**
   The parameter modifier indicates the communication function entity for which the RSVP parameters apply.
   Type: Alphanumeric value set.
   Range: UNI or ENNI.

Spec block parameters

1. **hellointerval** – Hello Interval
   This parameter is used to indicate the time between two succeeding HELLO messages to be sent to any neighbor in hello interworkings.
   Type: Numeric string.
   Range: 1 … 60, indicating the Hello Interval in seconds
2. **maxhelloretransmit** – Maximum Hello Retransmissions
   This parameter is used to indicate the number of Hello Intervals without any HELLO_REQUEST or HELLO_ACK object received from a neighbor after which communication to this neighbor is considered as lost.
   Type: Numeric string.
   Range: 1 to 10.
   Default: 3
3. **srefreshtimer** – Srefresh Timer
   This parameter is used to indicate the maximum time between two consecutive SREFRESH requests for a UNI or network connection towards affected neighbors in srefresh interworkings.
   Type: Numeric string.
   Range: 10 … 180 indicating the Srefresh time in seconds.
   Default: 15.

Example output

The following is an example **ED-RSVP-UNI** notification in case the *maxhelloretransmit* parameter has been changed:

```
   LT-UNITE 06-06-30 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=06-06-30:ED-RSVP-UNI:::maxhelloretransmit=8″
```

```
;
```

ED-STATE-EQPT (CP) notification

Output format

Syntax: **`ED-STATE-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of a circuit pack.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for any circuit pack
   - LoxcAID::=UnivSlotAID-loxc for LOXC pack in universal slots.

   The keyword **`usall`** is allowed for Simple_UnivSlotAID
   The keyword **`xcall`** is allowed for Simple_XCSlotAID
   The keyword **`ctlall`** is allowed for Simple_CTLSlotAID
   The keyword **`all`** is allowed for Simple_SlotAID
2. **`opstate`** – Operational State
   Each circuit pack has a retrievable attribute indicating whether circuit pack is operational for the system internal communication.
   Type: Alphanumeric value set.
   Range:
   - OS_ENABLED
   - OS_DISABLED (default)
   - OS_SYNCHING.

Example output

```
   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-14:ED-STATE-EQPT:1-1-#-#-1-cp:::opstate=OS_ENABLE”;
```

ED-STATE-EQPT (slot) notification

Output format

Syntax: **`ED-STATE-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of a slot.
   Type: AID
   Range:
   - SlotAID

   The keyword **`usall`** is allowed for Simple_UnivSlotAID.
   The keyword **`xcall`** is allowed for Simple_XCSlotAID.
   The keyword **`ctlall`** is allowed for Simple_CTLSlotAID.
   The keyword **`all`** is allowed for Simple_SlotAID, Simple_UnivSlotAID, Simple_XCSlotAID, Simple_CTLSlotAID.
   The keyword **`all`** is not allowed on bay/shelf level.

Spec block parameters

1. **`holdst`** – Holder Status
   Each provisionable slot has an attribute indicating its equipage state.
   Type: Alphanumeric value set.
   Range:
   - EMPTY. The system uses the value ″EMPTY″ if no circuit pack is equipped in that slot.
   - VALID. The system shall use the value ″Valid″ if a circuit pack is equipped in that slot as provisioned.
   - UNEXPECTED. The system shall use the value ″Unexpected″ if a different circuit pack is equipped in that slot as provisioned or if the circuit pack is not acceptable for this slot.
   - UNKNOWN. The system shall use the value ″Unknown″ if a circuit pack is equipped in that slot with non-readable functional type or illegal values for the functional type.

   Default: EMPTY.

Example output

```
    UNITE-NE 00-11-14 08:00:30
A   001 REPT DBCHG
    ″TIME=08-00-12,DATE=00-11-14:ED-STATE-EQPT:1-1-#-#-1::holdst=VALID″
```

```
;
```

ED-STATE-EQPT (socket) notification

Output format

Syntax: **`ED-STATE-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **`all`** is not allowed on bay/shelf level.

Spec block parameters

1. **`holdst`** – Holder Status
   Each provisionable socket shall have an attribute indicating its equipage state.
   Type: Alphanumeric value set.
   Range:
   - EMPTY, the system uses the value ″EMPTY″ if no module is equipped in that socket.
   - VALID, the system uses the value ″Valid″ if a module is equipped in that socket as provisioned.
   - UNEXPECTED, the system uses the value ″Unexpected″ if a different module is equipped in that socket as provisioned or if the module is not acceptable for this socket.
   - UNKNOWN, the system uses the value ″Unknown″ if a module is equipped in that socket with non-readable functional type or illegal values for the functional type.

   For further details w.r.t. state transitions for holder status – see SRD_Configuration-Management.
   Default: EMPTY.

Example output

This example shows an ED-STATE-EQPT notification reporting a plug out of a module from the socket:

```
UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-14:ED-STATE-EQPT:1-1-#-#-5-sc2::holdst=EMPTY″
;
```

ED-STATE-EQPT (system): Edit equipment state

The **ED-STATE-EQPT** command modifies the state information on system (NE) level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 4

Abortable: no

Input format

Command syntax: **ED-STATE-EQPT**:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system; the usage of the keyword **all** on system level is not allowed.
   Default: system
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **mcond** – Maintenance Condition
   This parameter denotes whether the system condition is in maintenance condition or not.
   Type: Alphanumeric value set.
   Range:
   - Y = maintenance mode
   - N = normal mode.

Output format

If the **ED-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

```
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the **RTRV-HDR** command.

Detailed behavior description

The command switches the system to the maintenance condition or back to normal condition. There is an autonomous system reset when the maintenance condition is left. The following attribute of the system entity is updated due to that operation:

mcond as specified in the command input.

The modification is reported by means of an ED-STATE-EQPT notification for the system.

Updated affected entities:

Setting *mcond* = Y causes that *opstate* = OS_DISABLED for all non-CTL circuit packs in the system. The change is reported by means of an ED-STATE-EQPT notification for the circuit pack.

Volatile maintenance condition

The maintenance condition is not stored in the configuration database. During reboot, all system states relevant for maintenance condition are re-evaluated to determine whether the system starts up in maintenance condition. As a consequence of this behavior, a maintenance condition established using ED-EQPT will be left after a reboot, if no other state causing a maintenance condition is present in the system (for example, a database serial number mismatch).

Example input/output

```
ED-STATE-EQPT:LT-UNITE:system:123456:::mcond=y;

   LT-UNITE 00-11-14 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ED-STATE-EQPT** command.

Related TL1 messages

“RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400)

ED-STATE-EQPT (system) notification

Output format

Syntax: **`ED-STATE-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system; the usage of the keyword **`all`** on system level is not allowed.
   Default: system

Spec block parameters

1. **`commitstate`** – SW Commitment Status
   The system provides an indication of the SW generic commitment status.
   Type: Alphanumeric value set.
   Range:
   - PENDING = the SW generic is in trial phase, switch back to previous generic possible
   - COMMITTED = the SW generic is comitted, no switch back possible.
2. **`dbstate`** – Data Base Change State
   The system provides an indication of the validity of the active data base.
   Type: Alphanumeric value set.
   Range:
   - EMPTY = data base state empty.
   - NEW = data base state new
   - VALID = database state valid
   - UNKNOWN = database state is not known.

   Default: UNKNOWN.
3. **`mcond`** – Maintenance Condition
   This parameter denotes whether the system condition is in maintenance condition or not.
   Type: Alphanumeric value set.
   Range:
   - Y = maintenance mode
   - N = normal mode.

Example output

```
ED-STATE-EQPT:LT-UNITE:system:123456:::mcond=n;
```

```
   LT-UNITE 00-11-14 08:00:00
M  123456 COMPLD
;

   LT-UNITE 00-11-14 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-14:ED-STATE-EQPT:system::mcond=N"
;
```

ED-STM1E: Edit STM1E port

The **ED-STM1E** command is used to provision all parameters related to the electrical STM1E port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM1E**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Access Identifier of an STM1E port.
   Type: AID.
   Range: SPortAID of STM1E port.
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   The setting of this parameter is allowed when the related STM1E port is created, it is used when *sdsfmode* = BURST.
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
   The setting of this parameter is allowed when the related STM1E port is created, it is used when *sdfsmode* = POISSON.

6. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
   The setting of this parameter is allowed when the related STM1E port is created, it is used when *sdfsmode* = POISSON.
7. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
8. **`femspm`** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port’s bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

11. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
12. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    This identifies the maximum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    maxNNXCRate is introduced specifically for use by ONNS application.
    Type: Alphanumeric value set.
    Range: VC4 (default), VC3.
13. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    This identifies the minimum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    minNNXCRate is introduced specifically for use by ONNS application.
    Type: Alphanumeric value set.
    Range: VC3 (default), VC4.
14. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

15. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

16. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

17. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

18. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
19. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP
20. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

21. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default value for SDH: STM1: 28800.

22. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default value for SDH: STM1: 28800.

23. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

24. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
25. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
26. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255.
27. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: 1 – 64000.
Default value for SDH: STM1: 2400.
28. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

29. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
30. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
31. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
32. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
33. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.

34. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
35. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
36. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
37. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (*strcwfmt* = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string, dependent on the *strcwfmt* parameter.

Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

38. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).
39. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
   - TR-only (lso known as the implicit reset method)
   - TR-RTR (lso known as the explicit reset method).

   Type: Alphanumeric value set.
Range: TR or TRRTR (default).
40. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Output format

If the **`ED-STM1E`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

;

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM1 notification for port (except for parameter *istd;* see below).

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc). When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term "PM monitor" has the same semantics as the term "PM package". The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*

- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Example input/output

```
ED-STM1E:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14,UNEQUIPOSIG=14;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM1E:1-1-#-#-1-1::inputsig=14,unequiposig=14:"
```

;

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-STM1E command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross-connections.

Error code:

```
SNVS – Status, Not in Valid State, signal structure cannot be changed
for connected parts of a fixed port
```

Validation rule: *unequiposig* consistent with existing cross-connections

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross-connections.

Error code:

```
SNVS – Status, Not in Valid State, signal structure cannot be changed
for connected parts of a fixed port
```

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

```
SNVS – Status, Not in Valid State, Timing reference is assigned to this
port
```

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

```
SNVS – Status, Not in Valid State, DCC channels are active for this port
```

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:

`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:
`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:
`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

```
SNVS – Status, Not in Valid State, Port must not be member of a
protection group
```

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

```
SNVS – Status, Not in Valid State, port mode change not allowed for
INNI ports
```

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:

```
SNVS – Status, Not in Valid State, J0 parameters can not be set for
INNI ports
```

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, `*`ndetmd`*` must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

`IDNV – Input, Data Not Valid, zero values can only be set for all neighbor information consistently`

ONNS EDGE port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following must be fulfilled:

- No cross-connections of xconcls = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters, if there still exist any cross-connections for the EDGE port, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

```
SNVS – Status, Not in Valid State, NN Cross-connections associated to
end-to-end network connections still existing
```

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* already assigned as *networkTNAAddrVCAT* to any VCG in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values, if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI port related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if UNI-N node is enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabeld`

Validation rule: Check of *UNI* parameters

It is not possible to change the *portClass* to UNI, unless valid values for the following UNI parameters are given:

clientNodeID

networkTNAType

networkTNAAddress

localLPID

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if *portcontrol* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a BLSR/ MS-SPRing protection group`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if there still exist any NN cross connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Change the *portClass* away from UNI

It is not possible to modify the parameter servcond to OUTOFSERVICE for UNI ports.

Error code:

`SNVS – Status, Not in Valid State, Cannot change servcond for this port class`

Validation rule: Some Parameter Values not settable for ports of *portClass* UNI

The following parameter is not changeable to value ADAPTIVE forvalue ADAPTIVE for ports with *portClass* UNI:

trbmd

Error code:

`SNVS – Status, Not in Valid State, can not change portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It shall not be possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to any EDGE port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means, that UNI and EDGE ports are not to be mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Related TL1 messages

"RTRV-STM1E: Retrieve STM1E port" (p. 3-2849)

References

GR-199-CORE (ED-*rr*)

ED-STM1E notification

Output format

The ED-STM1E notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Syntax: **`ED-STM1`**:*aid*::*spec_block*

Output parameter

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an STM1E port.
   Type: AID.
   Range: SPortAID of STM1E port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.
    Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU
11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port’s bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.
13. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this

attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

14. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).

15. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
This identifies the maximum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
maxNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC4 (default), VC3.

16. **`minNNXCRate`** – Minimum NN Cross Connect Rate
This identifies the minimum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
minNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC4, VC3 (default).

17. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

18. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

19. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

20. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

21. **`nemspm`** – MS Near-End PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
22. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.
    Range:
    - IPV4 (default)
    - IPV6
    - NSAP
23. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

24. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default value for SDH: STM1: 28800.

25. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default value for SDH: STM1: 28800.

26. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

27. **`pmrsneses`** – Performance Monitoring RS Near-End SES Threshold
This parameter describes the RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: 1 – 64000.
Default value for SDH: STM1: 2400.
28. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
29. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
30. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255.
31. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

32. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
33. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
34. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
35. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports of corresponding line rate (see definition of output parameter sonet in "RTRV-OC12: Retrieve OC12 port" (p. 3-2620)) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
36. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.

37. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
38. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.
39. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1)
    - 64 for 64 byte string (SONET).
40. **strcout** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter. If ITU/ETSI mode 2 is selected (*strcwfmt* = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string dependent on the *strcwfmt* parameter.

Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

41. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1)
    - 64 for 64 byte string (SONET).
42. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

43. **`trbmd`** – Tributary mode
This parameter is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in RTRV-EQPT (system) command.

44. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Example input/output

```
ED-STM1E:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14,UNEQUIPOSIG=14;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-STM1E:1-1-#-#-1-1::inputsig=14,unequiposig=14:″
;
```

ED-STM1E autonomous notifications

LOS for transmission port STM1E clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-STM1E notification with parameter *pmode*.

Change of neighbor Node and Neighbor Port Information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID* or *neighborPortID* is notified by the related ED-STM1E notification.

If ndetmd=AUTO a change of the attribute *neighborASTNNodeID* is notified by the related ED-STM1E notification.

ED-STM1: Edit STM1 port

The command **ED-STM1** is used to provision all parameters related to optical STM1 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM1**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   Type: AID.
   Range: SPortAID of STM1 port.
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   The setting of this parameter is allowed when the related STM1 port is created, it is used when *sdfsmode* = BURST.
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   The setting of this parameter is allowed when the related STM1 port is created, it is used when *sdfsmode* = BURST.
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
   The setting of this parameter is allowed when the related STM1 port is created, it is used when *sdfsmode* = POISSON.

6. **dexcthr** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
   The setting of this parameter is allowed when the related STM1 port is created, it is used when *sdfsmode* = POISSON.
7. **exprstrc** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
8. **femspm** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **frcdus** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
10. **inputsig** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.
11. **istd** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.

Range:

- SONET
- SDH.

Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 or VC4 (default)

16. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default) or VC4.

17. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

18. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
19. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″
20. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

21. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
22. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
23. **`nemspm`** – MS Near-End PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
24. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

25. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

26. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET:

    - STM1: 28800
    - OC3: 154.

27. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

28. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

29. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

30. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

31. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0 (default) …255. The unit of this value is ″milliseconds″.

32. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

33. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it

can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

34. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

35. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

36. **rspm** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

37. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.

Range:

- ENABLE (default): enable consequent action for TIM
- DISABLE: disable consequent action for TIM.

38. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
39. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
40. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
41. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
42. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

43. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
44. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the *strcwfmt* parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

45. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
46. **tcamd** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).
47. **unequiposig** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Output format

If the ED-STM1 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM1 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SDH to SONET has the following impact related to fault management:

- all SDH alarms related to that port are cleared
- the corresponding SONET alarms are raised.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc). When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

Switching of interface standard:

The switching of istd should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *femspm* is mapped to *felpm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SONET
- *nemspm* is mapped to *nelpm* (no changes)
- *rspm* is mapped to *spm* (no changes)
- *rstimca* is set to the default value for a SONET port
- *rstimmon* is set to the default value for a SONET port
- *sdhpn* is mapped to SONET (no changes)
- *sdsfmode* is set to the default value for a SONET port
- *sdthr* is set to the default value for a SONET port
- *sfthr* is set to the default value for a SONET port
- *trbmd* is set to the default value for a SONET port
- *unequiposig* is set to its default value
- *z0outmode* is set to its default value for a SONET port.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *sltca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmmsfeses* is set to the default value for a SONET port
- *pmmsneses* is set to the default value for a SONET port
- *pmrsneses* is set to the default value for a SONET port.

Switching of interface standard IV:

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *minNNXCrate* is set to the default value for a SONET tributary
- *maxNNXCrate* is set to the default value for a SONET tributary

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value for a SONET tributary
- *brstthr* is set to the default value for a SONET tributary
- *pse* (PDI-P parameter) is set to the default value for a SONET tributary
- *sdsfmode* is set to the default value for a SONET tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *degthr* is set to the default value for an SONET tributary
- *dexcthr* is set to the default value for an SONET tributary
- *ptca* is set to the default value for a SONET tributary
- *tcamd* is set to the default value for a SONET tributary.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmhpneses* is set to the default value for an SONET tributary
- *pmhpfeses* is set to the default value for an SONET tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-STMn command all resulting parameter changes are reported by ED-OCn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-STMn command all resulting tributary parameter changes are reported by ED-STS1 notifications.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard *istd* to SONET includes an change of *tcamd* from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*

- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

The following parameters of the ENNI parameter set are mandatory and is set different to their default values:

- *remAdvNodeID (if a corresponding level 1routing model is defined)*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

This is covered by corresponding validation rules.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-STM1:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14,UNEQUIPOSIG=14;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM1:1-1-#-#-1-1::inputsig=14,unequiposig=14:"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the ED-STM1 command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connections are still on this port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd* change

Check that the interface standard parameter (*istd*) is not changed when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no protection for *istd* change

Check that the interface standard parameter (*istd*) is not changed when this port is in an protection group.

Error code:
`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no test access session for *istd* change

Check that *istd* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

```
SNVS – Status, Not in Valid State, LO cross-connections, substructure
or conversion cross-connections still exist
```

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*

- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, ndetmd must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

`IDNV – Input, Data Not Valid, zero values can only be set for all neighbor information consistently`

ONNS EDGE port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following must be fulfilled:

- No cross-connections of xconcls = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters, if there still exist any cross-connections for the EDGE port, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means, that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* already assigned as *networkTNAAddrVCAT* to any VCG in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values, if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI port related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:
`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:
`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1+1 MSP/APS group.

Error code:
`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1:1 MSP/APS group.

Error code:
`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameter

It is not possible to change the *portClass* to UNI, unless valid values for the following UNI parameters are given:

- *clientNodeI*
- *networkTNATyp*
- *networkTNAAddres*
- *localLPID*

Error code:
`IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT value`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of *portClass* UNI

The following parameter is not changeable to the value ADAPTIVE for ports with *portClass* UNI:

trbmd

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*

- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It is not possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned to any EDGE port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:
`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if cross-connections on that port are still existing.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:
`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections are still existing`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:
`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if cross connect loopback exists for any tributary related to this port.

Error code:
`SNVS – Status, Not in Valid State, cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if the port is any member of a 1:1 MSP/APS group.

Error code:
`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI, unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*

- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is unique within the system.

Error code:

`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`

Related TL1 messages

"RTRV-STM1: Retrieve STM1 port" (p. 3-2863)

References

GR-199-CORE (ED-*rr*)

ED-STM1 autonomous notifications

LOS for transmission port STM1 clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-STM1 notification with parameter *pmode*.

Change of neighbor node and neighbor port Information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID* or *neighborPortID* is notified by the related ED-STM1 notification.

If ndetmd=AUTO a change of the attribute *neighborASTNNodeID* is notified by the related ED-STM1 notification.

ED-STM1 notification

Output format

The **ED-STM1** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ED-STM1**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID.
   Range: SPortAID of STM1 port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter inputsig showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-… N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.

6. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

7. **`dexcthr`** – Excessive Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

8. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU
11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.
13. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET.
Type: Alphanumeric value set.
Range:
    - SONET
    - SDH.

    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
14. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

16. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

17. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.

Range:

- YES
- NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 or VC4 (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default) or VC4.

20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This

parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0.

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
   - IPV4 (default)
   - IPV6
   - NSAP

28. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
  0000000000000000000000000000000000000000…
  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET:

    - STM1: 28800
    - OC3: 154.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

31. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM1: 2400
    - OC3: 155.
33. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
34. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
35. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.
36. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0
If *portClass* has a value supporting this parameter, this default value shall have the meaning of “no ID provisioned”.

37. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

38. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

39. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: ″″

40. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

41. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

42. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

43. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

44. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

45. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
- BURST (default)
- POISSON.

46. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
- INSERVICE (default): Provisioning values for service are accepted
- OUTOFSERVICE: Provisioning values for service are not accepted.

47. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
- 1 for specific repeating byte (SONET)
- 2 (default)for non-specific repeating byte (ITU/ETSI mode 2)

- 16 for 16 byte specific string (ITU/ETSI mode 1). 64 for 64 byte string (SONET)
- 16 for 16 byte specific string (ITU/ETSI mode 1). 64 for 64 byte string (SONET)

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter. If ITU/ETSI mode 2 is selected (*strcwfmt* = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default)for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1). 64 for 64 byte string (SONET)
    - 16 for 16 byte specific string (ITU/ETSI mode 1). 64 for 64 byte string (SONET)

51. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).

52. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
    Type: Alphanumeric value set.
    Range: ADAPTIVE, FIXED
    Default: The initial value is defined as the trbmddflt parameter on system level, see definition in RTRV-EQPT (system) command.
53. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

Example output

```
ED-STM1:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14,UNEQUIPOSIG=14;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM1:1-1-#-#-1-1::inputsig=14,unequiposig=14:"
;
```

ED-STM4: Edit STM4 port

The command **ED-STM4** is used to provision all parameters related to optical STM4 ports.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM4**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Type: AID.
   Range: SPortAID of STM4 port.
   For changing the portControl parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   The setting of this parameter is allowed when the related STM4 port is created, it is used when *sdfsmode* = BURST.
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   The setting of this parameter is allowed when the related STM4 port is created, it is used when *sdfsmode* = BURST.
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
   The setting of this parameter is allowed when the related STM4 port is created, it is used when *sdfsmode* = POISSON.

6. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
   The setting of this parameter is allowed when the related STM4 port is created, it is used when *sdfsmode* = POISSON.
7. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
8. **`femspm`** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C.
    Default: 44.
11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.

Range:

- SONET
- SDH.

Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

12. **linkResCls** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
13. **localLinkID** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **localLPID** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4 or VC44C (default)

16. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4 or VC44C.

17. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

18. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
19. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″
20. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

21. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
22. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
23. **`nemspm`** – MS Near-End PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
24. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

25. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

26. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:

    - STM4: 115200
    - OC12: 615.

27. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

28. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

29. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
   - STM4: 2400
   - OC12: 616.
30. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
31. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
32. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0 (default) …255. The unit of this value is ″milliseconds″.
33. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

34. **remLinkID** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

35. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

36. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

37. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
38. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
39. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
40. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
41. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.

42. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.
43. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
44. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
45. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

46. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex
47. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
48. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).
49. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port’s bandwidth. It determines the rate(s) of the “unequipped” signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4 or 4c representing VC3, VC4 or VC4-4C.
Default: 44.

Output format

If the ED-STM4 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM16 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SDH to SONET has the following impact related to fault management:

- all SDH alarms related to that port are cleared
- the corresponding SONET alarms are raised.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

Switching of interface standard.

The switching of istd should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *femspm* is mapped to *felpm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SONET
- *nemspm* is mapped to *nelpm* (no changes)
- *rspm* is mapped to *spm* (no changes)
- *rstimca* is set to the default value for a SONET port
- *rstimmon* is set to the default value for a SONET port
- *sdhpn* is mapped to SONET (no changes)
- *sdsfmode* is set to the default value for a SONET port
- *sdthr* is set to the default value for a SONET port
- *sfthr* is set to the default value for a SONET port
- *trbmd* is set to the default value for a SONET port
- *unequiposig* is set to its default value
- *z0outmode* is set to its default value for a SONET port.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *sltca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmmsfeses* is set to the default value for a SONET port
- *pmmsneses* is set to the default value for a SONET port
- *pmrsneses* is set to the default value for a SONET port.

Switching of interface standard IV:

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *minNNXCrate* is set to the default value for a SONET tributary
- *maxNNXCrate* is set to the default value for a SONET tributary

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value for a SONET tributary
- *brstthr* is set to the default value for a SONET tributary
- *pse* (PDI-P parameter) is set to the default value for a SONET tributary
- *sdsfmode* is set to the default value for a SONET tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *degthr* is set to the default value for an SONET tributary
- *dexcthr* is set to the default value for an SONET tributary
- *ptca* is set to the default value for a SONET tributary
- *tcamd* is set to the default value for a SONET tributary.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmhpneses* is set to the default value for an SONET tributary
- *pmhpfeses* is set to the default value for an SONET tributary.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-STMn command all resulting parameter changes are reported by ED-OCn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-STMn command all resulting tributary parameter changes are reported by ED-STS1 notifications.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard *istd* to SONET includes an change of *tcamd* from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when *portClass* is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*

- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters values different to their default values are mandatory.

This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-STM4:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14-33-24,UNEQUIPOSIG=14-33-24;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-STM4:1-1-#-#-1-1::inputsig=14-33-
      24,unequiposig=14-33-24:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-STM4 command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd* change

Check that the interface standard parameter (*istd*) is not changed when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no protection for *istd* change

Check that the interface standard parameter (*istd*) is not changed when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no test access session for *istd* change

Check that *istd* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:

`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:
`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:
`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:
`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*

- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State,` *`ndetmd`* `must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

```
IDNV – Input, Data Not Valid, zero values can only be set for all
neighbor information consistently
```

ONNS EDGE port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following must be fulfilled:

- No cross-connections of xconcls = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters, if there still exist any cross-connections for the EDGE port, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means, that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* already assigned as *networkTNAAddrVCAT* to any VCG in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values, if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI port related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameter

It is not possible to change the *portClass* to UNI, unless valid values for the following UNI parameters are given:

- *clientNodeI*
- *networkTNATyp*
- *networkTNAAddres*
- *localLPID*

Error code:

`IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT value`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of *portClass* UNI

The following parameter is not changeable to the value ADAPTIVE for ports with *portClass* UNI:

trbmd

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It is not possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned to any EDGE port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections are still existing`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if the port is any member of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI, unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*

- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:

`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:

```
SNVS – Status, Not in Valid State, can not change servcond for this
port class
```

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:

```
SNVS – Status, Not in Valid State, only allowed for ENNI port in
lockout state
```

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

```
SNVS – Status, Not in Valid State, Cross-connections still existing
```

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is unique within the system.

Error code:
`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:
`IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`

Related TL1 messages

"RTRV-STM4: Retrieve STM4 port" (p. 3-2881)

References

GR-199-CORE (ED-*rr*)

ED-STM4 autonomous notifications

LOS for transmission port STM4 clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-STM4 notification with parameter *pmode*.

Change of neighbor node and neighbor port Information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-STM4 notification.

ED-STM4 notification

Output format

The ED-STM4 notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ED-STM4**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
Type: AID.
Range: SPortAID of STM4 port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range: 2 to 10.
Default: 7.
3. **brstthr** – Bursty Error Threshold
Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range (as percentage): 5, 10, … 50.
Default: 30.
4. **clientNodeID** – Client Node IDentifier
The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.

6. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

7. **`dexcthr`** – Excessive Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

8. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.
    Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU
11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C.
    Default: 44.
13. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range:
    - SONET
    - SDH.

    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
14. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute

represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.

Range:

- YES
- NO (default).

18. **maxNNXCRate** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4 or VC44C (default)

19. **minNNXCRate** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4 or VC44C.

20. **ndetmd** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **neighborASTNNodeID** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This

parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **neighborNodeID** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **neighborPortID** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **neighborSigPCID** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:

    - STM4: 115200
    - OC12: 615.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

31. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 2400
    - OC12: 616.

33. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

34. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

35. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.

36. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

38. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

39. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

40. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
41. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
43. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
45. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.

Range:

- BURST (default)
- POISSON.

46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1)
    - 64 for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter. f ITU/ETSI mode 2 is selected (*strcwfmt* = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1 for specific repeating byte (SONET)
    - 2 (default) for non-specific repeating byte (ITU/ETSI mode 2)
    - 16 for 16 byte specific string (ITU/ETSI mode 1)
    - 64 for 64 byte string (SONET).
51. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

52. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in RTRV-EQPT (system) command.

53. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4 or 4c representing VC3, VC4 or VC4-4C.
Default: 44.

Example output

```
ED-STM4:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=14-33-24,UNEQUIPOSIG=14-33-24;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM4:1-1-#-#-1-1::inputsig=14-33-
      24,unequiposig=14-33-24:"
;
```

ED-STM16: Edit STM16 port

The command **ED-STM16** is used to provision all parameters related to optical STM16 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM16**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Description: Access Identifier of an STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **all** is allowed up to shelf level.
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
The setting of this parameter is allowed when the related STM16 port is created, it is used when *sdfsmode* = BURST.
Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range: 2 to 10.
Default: 7.

3. **`brstthr`** – Bursty Error Threshold
The setting of this parameter is allowed when the related STM16 port is created, it is used when *sdfsmode* = BURST.
Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range (as percentage): 5, 10, … 50.
Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
The setting of this parameter is allowed when the related STM16 port is created, it is used when *sdfsmode* = POISSON.

6. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
   The setting of this parameter is allowed when the related STM16 port is created, it is used when *sdfsmode* = POISSON.
7. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
8. **`femspm`** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port’s bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.

Default: The initial value of X is 4.
Example:
An STM-16 signal might be represented by 93-14-14C-243 which implies 9 VC3s, 1 VC4, 1 VC4-4C, 24 VC3s. The representation is not unique, for example it could also be 33-63-14-14C-243. Worst case of string length for STM16: 47 characters.

11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range:
    - SONET
    - SDH.

    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C or VC416C (default)

16. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C or VC416C.

17. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

18. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

19. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

20. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

21. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

22. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

24. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this

attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

25. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

26. `pmmsfeses` – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-16/OC-48: 1 – 3072000
   - STM-64/OC-192: 1 – 12288000
   - STM-256/OC-768: 1 – 49152000.

   Default: The default values differ for SDH and SONET and the different rates:
   - STM16: 460800
   - STM64: 1843200
   - STM256: 7372800
   - SONET OC48: 2459
   - SONET OC192: 9835
   - SONET OC768: 39340.
27. `pmmsneses` – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-16/OC-48: 1 – 3072000
   - STM-64/OC-192: 1 – 12288000
   - STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

28. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
29. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
30. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

31. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
32. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.
33. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
34. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

35. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

36. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

37. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

38. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

39. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.

Range:

- ENABLE (default): enable consequent action for TIM
- DISABLE: disable consequent action for TIM.

40. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
41. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
42. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
43. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
44. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

45. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
46. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex

47. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
48. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).
49. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* shall not need the identical setting. Mixed rates for tx and rx side of the port is allowed.
Type: numeric value set
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
Default: 164 = all VC4.

Output format

If the ED-STM16 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM16 notification for port (except for parameter *istd;* see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SDH to SONET has the following impact related to fault management:

- all SDH alarms related to that port are cleared
- the corresponding SONET alarms are raised.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value, the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *femspm* is mapped to *felpm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SONET
- *nemspm* is mapped to *nelpm* (no changes)
- *rspm* is mapped to *spm* (no changes)
- *rstimca* is set to the default value for a SONET port
- *rstimmon* is set to the default value for a SONET port
- *sdhpn* is mapped to SONET (no changes)
- *sdsfmode* is set to the default value for a SONET port
- *sdthr* is set to the default value for a SONET port
- *sfthr* is set to the default value for a SONET port
- *trbmd* is set to the default value for a SONET port
- *unequiposig* is set to its default value
- *z0outmode* is set to its default value for a SONET port.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *sltca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port.

The attribute *tcamd* does not exist for SONET ports. SONET ports only support TR-only mode. Thus the mode must be internally set to TR-only.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmmsfeses* is set to the default value for a SONET port
- *pmmsneses* is set to the default value for a SONET port
- *pmrsneses* is set to the default value for a SONET port.

Switching of interface standard IV:

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *minNNXCrate* is set to the default value for a SONET tributary
- *maxNNXCrate* is set to the default value for a SONET tributary

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value for a SONET tributary
- *brstthr* is set to the default value for a SONET tributary
- *pse* (PDI-P parameter) is set to the default value for a SONET tributary
- *sdsfmode* is set to the default value for a SONET tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *degthr* is set to the default value for an SONET tributary
- *dexcthr* is set to the default value for an SONET tributary
- *ptca* is set to the default value for a SONET tributary
- *tcamd* is set to the default value for a SONET tributary.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmhpneses* is set to the default value for an SONET tributary
- *pmhpfeses* is set to the default value for an SONET tributary.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-STMn command all resulting parameter changes are reported by ED-OCn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-STMn command all resulting tributary parameter changes are reported by ED-STS1 notifications.

For efficiency reasons the ED-STS1 notifications are sent out as a single notification using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to

- 1 – 3, in case of an STM1/OC3 port
- 1 – 12, in case of an STM4/OC12 port
- 1 – 48, in case of an STM16/OC48 port
- 1 – 192, in case of an STM64/OC192 port
- 1 – 768, in case of an STM256/OC768 port.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard *istd* to SONET includes an change of *tcamd* from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*

- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters, values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-STM16:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM16:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-STM16 command.

Validation rule: Enough bandwidth reserved for 2F protection group

Check of the tributary rate (inputsig/unequiposig) if the port is member of a 2f protection group. Then the assigned bandwith is max. half of the port’s bandwith.

Error code:
`IDNV – Input, Data Not Valid, Assigned bandwidth exceeds half port’s bandwith`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd* change

Check that the interface standard parameter (*istd*) is not changed when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no protection for *istd* change

Check that the interface standard parameter (*istd*) is not changed when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no test access session for *istd* change

Check that *istd* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **ED** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **ED** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:
`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:

`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

```
SNVS – Status, Not in Valid State, port mode change not allowed for
INNI ports
```

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:

```
SNVS – Status, Not in Valid State, J0 parameters can not be set for
INNI ports
```

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

```
SNVS – Status, Not in Valid State, only allowed for INNI port in
lockout state
```

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State, ndetmd must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

`IDNV – Input, Data Not Valid, zero values can only be set for all neighbor information consistently`

ONNS EDGE port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following must be fulfilled:

- No cross-connections of xconcls = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters, if there still exist any cross-connections for the EDGE port, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means, that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* already assigned as *networkTNAAddrVCAT* to any VCG in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values, if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI port related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameter

It is not possible to change the *portClass* to UNI, unless valid values for the following UNI parameters are given:

- *clientNodeI*
- *networkTNATyp*
- *networkTNAAddres*
- *localLPID*

Error code:

`IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT value`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of *portClass* UNI

The following parameter is not changeable to the value ADAPTIVE for ports with *portClass* UNI:

trbmd

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It is not possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned to any EDGE port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections are still existing`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if the port is any member of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI, unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
```
SNVS – Status, Not in Valid State, can not change servcond for this
port class
```

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*

- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:
`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is unique within the system.

Error code:
`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

```
IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to
their DEFAULT values
```

Related TL1 messages

"RTRV-STM16: Retrieve STM16 port" (p. 3-2899)

References

GR-199-CORE (ED-*rr*)

ED-STM16 autonomous notifications

LOS for transmission port STM16 clears

If LOS is cleared and current port mode is AUTO, then *pmode* is updated (new value: MON). This update is reported via ED-STM16 notification with parameter *pmode*.

Change of neighbor node and neighbor port Information

If ndetmd=AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-STM16 notification.

ED-STM16 notification

Output format

The **ED-STM16** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ED-STM16**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Description: Access Identifier of an STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (trbmd = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List .
   Type: numeric value set
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **`femspm`** – Far-end Multiplex Section Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **frcdus** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.

    Range:

    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU

11. **fsflpbkstat** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

12. **inputsig** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
    Default: The initial value of X is 4.
    Example:
    An STM-16 signal might be represented by 93-14-14C-243 which implies 9 VC3s, 1 VC4, 1 VC4-4C, 24 VC3s. The representation is not unique, for example it could also be 33-63-14-14C-243. Worst case of string length for STM16: 47 characters.

13. **istd** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.

    Range:

    - SONET
    - SDH.

    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3, VC4, VC44C or VC416C (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3 (default), VC4, VC44C or VC416C.

20. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **neighborASTNNodeID** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **neighborNodeID** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **neighborPortID** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this

attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

31. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

32. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.

33. **portClass** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

34. **`portControl`** – Port Control (PortAdministrative Mode)
 The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
 Type: Alphanumeric value set.
 Range: NORMAL (default), LOCKOUT

35. **`propagationDelay`** – Propagation Delay
 This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
 propagationDelay is used by the ONNS application.
 Type: Numeric string.
 Range: 0(default) …255. The unit of this value is ″milliseconds″.

36. **`remAdvNodeID`** – Remote Advertised Node IDentity
 The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
 For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
 This parameter is supported by ports of *portClass* ENNI.
 Type: Four dot separated numeric string
 Range: Each numeric string: 0…255
 Default: 0.0.0.0

37. **`remLinkID`** – Remote Link IDentity
 The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
 This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
 This parameter is supported by ports of *portClass* ENNI.
 Type: numeric value set
 Range: 0 (default) to 4294967295

38. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
39. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
40. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
41. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

43. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor
44. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
45. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
   - BURST (default)
   - POISSON.
46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.

This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
Type: Numeric string.
Default: 2

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
Type: Hexadecimal digits.
Range:
   - 1 byte string
   - 16 bytes string
   - 64 bytes string dependent on the strcwfmt parameter.

   Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
Type: Numeric string.
Default: 2

51. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
   - TR-only (lso known as the implicit reset method)
   - TR-RTR (lso known as the explicit reset method).

   Type: Alphanumeric value set.
Range: TR or TRRTR (default).

52. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in RTRV-EQPT (system) command.

53. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* shall not need the identical setting. Mixed rates for tx and rx side of the port is allowed.
Type: numeric value set
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
Default: 164 = all VC4.

Example output

```
ED-STM16:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=74-33-84,UNEQUIPOSIG=74-33-84;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-STM16:1-1-#-#-1-1::inputsig=74-33-
      84,unequiposig=74-33-84:″
;
```

ED-STM16T: Edit STM16T port

The command **ED-STM16T** is used to provision all parameters related to optical STM16T port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM16T**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
aid – Access Identifier
Access Identifier of STM16T port.
Type: AID.
Range: SPortAID of STM16T port.
The keyword **all** is not allowed.
3. **ctag** – Correlation Tag
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **exprstrc** – Value of Expected J0
For ED-STM16T this parametr refers to the incoming (inress) direction.
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.

Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
3. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SDH.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
4. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
5. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392

- SONET OC192: 8554
- SONET OC768: 22776.

6. **`pmrsnesesout`** – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
7. **`rsmstca`** – RS/MS TCA Profile Pointer
   Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
   Type: Alphanumeric string.
   Range: String with maximum of 24 characters.
   Default: DEFAULT0.
8. **`rspm`** – RS PM Enable
   For ED-STM16T this parameter refers to the incoming (ingress) direction.
   Enable or disable regenerator section (section) performance monitoring.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`rspmout`** – RS PM Enable in Outgoing Direction
   Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
10. **`rstimmon`** – RS TIM Monitor Enable
    For ED-STM16T this parameter refers to the incoming (Ingress) direction.
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.

Range:

- ENABLE: enable TIM Monitor
- DISABLE (default): disable TIM Monitor

11. **rstimmon**out – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor
12. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
13. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
14. **strcfmt** – Format of incoming J0
For ED-STM16T this parameter refers to the incoming (Ingress) direction.
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
15. **strcfmt**out – Format of outgoing J0
The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
Type: Numeric string.

Range:

- 1, for specific repeating byte (SONET)
- 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
- 16, for 16 byte specific string (ITU/ETSI mode 1)
- 64, for 64 byte string (SONET).

16. **`tcamd`** – TCA mode

    This parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable:

    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.

    Range: TR or TRRTR (default).

Output format

If the ED-STM16T command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM16T notification for port (except for parameter *istd;* see below).

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt/strcfmtout) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc/expstrcout) and the currently received J0 (strc/strcout).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-STM16T* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SDH to SONET has the following impact related to fault management:

- all SDH alarms related to that port are cleared
- the corresponding SONET alarms are raised.

Switching of interface standard – Impact on port parameters

Switching of interface standard:

The switching of *istd* should not result in a deletion and recreation of the port. NE internally there are no differences between SDH and SONET. In detail the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OC48T command for SONET parameter description):

- *istd* is set to SONET
- *rspm* is mapped to *spm* (no changes)
- *rspmout* is mapped to spmout (no changes)
- *pmrsneses* is set to the default value for a SONET port
- *pmrsnesesout* is set to the default value for a SONET port
- *rstimmon* is set to the default value for a SONET port
- *rstimmonout* is set to the default value for a SONET port
- *sdhpn* is mapped to SONET (no changes)
- *sltca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port.

All other port parameters remain unchanged.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-STM16T command all resulting parameter changes are reported by ED-OC48T notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard *istd* to SONET includes an change of *tcamd* from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Example input/output

```
ED-STM16T:UNITE-NE:1-1-#-#-1-1:123456:::RSPM=ENABLE,RSPMOUT=ENABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ED-STM16T:1-1-#-#-1-1::rspm=ENABLE,rspmout=ENABLE:"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-STM16T** command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:
`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:
`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **ED** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **ED** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Check CCB against *ptype* and *istd*

Check wether the CCB fits to *ptype* and *istd*.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Related TL1 messages

"RTRV-STM16T: Retrieve STM16T port" (p. 3-2917)

References

GR-199-CORE (ED-*rr*)

ED-STM16T notification

Output format

The **ED-STM16T** notification uses the general format of a REPT DBCHG notification.

Syntax

ED-STM16T*:aid::spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **all** is not allowed.

Spec block parameters

1. **exprstrc** – Value of Expected J0
   For ED-STM16T Notification this parameter refers to the incoming (inress) direction
   Expected regenerator section trace identifier. TThe parameter expstrc specifying the expected path trace is only relevant for the section trace supervision when the related trace read format strcfmt is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **exprstrcout** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction. The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.

3. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

4. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter *istddflt*.
   Type: Alphanumeric value set.
   Range: SDH
   Default: As specified via istddflt on system level, see definition in RTRV-EQPT (system).

5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

7. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
8. **`pmrsneses`**out – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
9. **`rsmstca`** – RS/MS TCA Profile Pointer
   Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
   Type: Alphanumeric string.
   Range: String with maximum of 24 characters.
   Default: DEFAULT0.
10. **`rspm`** – RS PM Enable
   For ED-STM16T this parameter refers to the incoming (ingress) direction.
   Enable or disable regenerator section (section) performance monitoring.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).

11. **`rspmout`** – RS PM Enable in Outgoing Direction
Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).

12. **`rstimmon`** – RS TIM Monitor Enable
For ED-STM16T Notification this parameter refers to the incoming (Ingress) direction.
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor

13. **`rstimmonout`** – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor

14. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in “RTRV-OC48: Retrieve OC48 port” (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

15. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

16. **`strcfmt`** – Format of incoming J0
    For ED-STM16T this parameter refers to the incoming (Ingress) direction.
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.

    Range:

    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
17. **`strcfmt`**out – Format of outgoing J0
    The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
    Type: Numeric string.

    Range:

    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).
18. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable:

    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

Example output

```
ED-STM16T:UNITE-NE:1-1-#-#-1-1:123456:::RSPM=ENABLE,RSPMOUT=ENABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-STM16:1-1-#-#-1-1::rspm=ENABLE,rspmout=ENABLE:”
;
```

ED-STM64: Edit STM64 port

The command **ED-STM64** is used to provision all parameters related to optical STM64 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STM64**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an STM64 port.
   Type: AID.
   Range: STM-64 port AID
   For changing the *portControl* parameter the ALL qualifier for the port aid is allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535

2. **`brstintvl`**– Bursty Interval
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = BURST.
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = BURST.
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = POISSON.
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.

6. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = POISSON.
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
7. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
8. **`fec`** – FEC Enable
   Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
   Type: Alphanumeric value set.
   Range:
   DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768 / STM256 basis low cost intra office 1310 nm optics.

   ENABLE / DISABLE for:
   - OC192 / STM64 OLS800G compatible 80 colour optics
   - OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
   - OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

   ENABLE (only) for:
   - OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G.

   Default:

DISABLE for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768 / STM256 basis low cost intra office 1310 nm optics.

ENABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

9. **`femspm`** – Far-end Multiplex Section Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).
10. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
    - ENABLE: (set a forced DUS/DNU)
    - DISABLE (default): (clear a forced DUS/DNU).
11. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
Default: 644.
12. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET.
Type: Alphanumeric value set.

Range:

- SONET
- SDH.

Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

13. **linkResCls** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

14. **localLinkID** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

15. **localLPID** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C, VC416C or VC464C (default)

17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C, VC416C or VC464C.

18. **`msrei`** – MS-REI mode
Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
Type: Alphanumeric value set.
Range:
    - SINGLE
    - EXTENDED (default).

19. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

20. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

21. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

22. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

23. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

26. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this

attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

27. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

28. **och** – optical channel enable
Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don’t have any impact on transmission.
Type: Alphanumeric value set.
Range:
    - for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
    - or ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
    - or ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
    - for client ports on packs which support OTN: DISABLE: disable, optical channel not applicable

    Default:
    - for OP10 and OP40 packs, which cannot support OCH: DISABLE
    - for OP10 packs, which cannot support OCH: ENABLE
    - for OP10 and OP40 packs, which only support OCH: ENABLE
    - for client ports on packs which support OTN: DISABLE.
29. **pmmsfeses** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200

- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

31. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

    Default:
    - AUTO: for optical SDH/SONET ports
    - NMON for client ports.

32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
33. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.
34. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
35. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
36. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

37. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

38. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

39. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

40. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

41. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

42. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

43. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

44. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

45. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC192: Retrieve OC192 port" (p. 3-2664)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

46. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
   - BURST (default)
   - POISSON.

47. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

48. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

49. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
50. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
51. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
52. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (lso known as the implicit reset method)
- TR-RTR (lso known as the explicit reset method).

Type: Alphanumeric value set.
Range: TR or TRRTR (default).

53. **unequiposig** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
Default: 644 = all VC4.

Output format

If the ED-STM64 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior

The command modifies the attributes (listed in the *spec_block* parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an ED-STM64 notification for port (except for parameter *istd*; see below).

Switching of interface standard – Ignore other parameters

If additional parameters beside *istd* are specified with the *ED-rr* command they will be ignored by the NE.

This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard – FM impact

The change of the interface standard of a port from SDH to SONET has the following impact related to fault management:

- all SDH alarms related to that port are cleared
- the corresponding SONET alarms are raised.

Supply outgoing J0 trace value and format together

When changing the format of the outgoing J0 trace *strcwfmt*, the parameter *strcout* specifying the outgoing J0 trace is always supplied within the same TL1-command to always have a consistent set of trace format and value.

When the format of the outgoing J0 trace *strcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J0 trace (mandatory parameter strcout) specified in the command is ignored.

Supply outgoing J0 trace value according to format of outgoing J0

When changing the outgoing J0 trace value , the format must be according to the format parameter *strcwfmt* for the outgoing J0 trace.

Change of J0 expected format - FM impact

When the expected J0 format (strcfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J0 (expstrc) and the currently received J0 (strc).

When the expected J0 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Switching of interface standard – Impact on port parameters

Switching of interface standard:

The switching of istd should not result in an deletion and recreation of the port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I:

In detail the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value
- *brstthr* is set to the default value
- *crstat* may change according to its semantics (see description of *crstat*)
- *femspm* is mapped to *felpm* (no changes)
- *inputsig* is set to the default value
- *istd* is set to SONET
- *nemspm* is mapped to *nelpm* (no changes)

- *rspm* is mapped to *spm* (no changes)
- *rstimca* is set to the default value for a SONET port
- *rstimmon* is set to the default value for a SONET port
- *sdhpn* is mapped to SONET (no changes)
- *sdsfmode* is set to the default value for a SONET port
- *sdthr* is set to the default value for a SONET port
- *sfthr* is set to the default value for a SONET port
- *trbmd* is set to the default value for a SONET port
- *unequiposig* is set to its default value
- *z0outmode* is set to its default value for a SONET port.

All other port parameters remain unchanged.

Switching of interface standard II:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *sltca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port.

The attribute *tcamd* does not exist for SONET ports. SONET ports only support TR-only mode. Thus the mode must be internally set to TR-only.

Switching of interface standard III:

In addition to the list of attributes above the following attributes are influenced by switching of the *istd* from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmmsfeses* is set to the default value for a SONET port
- *pmmsneses* is set to the default value for a SONET port
- *pmrsneses* is set to the default value for a SONET port.

Switching of interface standard IV:

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *minNNXCrate* is set to the default value for a SONET tributary
- *maxNNXCrate* is set to the default value for a SONET tributary

Switching of interface standard – Side effects on tributaries

The switching of *istd* should not result in an deletion and recreation of the tributaries of a port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SDH to SONET (see also the related RTRV-OC*n* command for SONET parameter description):

- *brstintvl* is set to the default value for a SONET tributary
- *brstthr* is set to the default value for a SONET tributary
- *pse* (PDI-P parameter) is set to the default value for a SONET tributary
- *sdsfmode* is set to the default value for a SONET tributary.

Switching of interface standard – Side effects on tributaries II

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *degthr* is set to the default value for an SONET tributary
- *dexcthr* is set to the default value for an SONET tributary
- *ptca* is set to the default value for a SONET tributary
- *tcamd* is set to the default value for a SONET tributary.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – Side effects on tributaries III

In addition to the list of attributes above the following side-effects apply to the related tributary attributes when switching the istd from SDH to SONET (see also the related RTRV-OCn command for SONET parameter description):

- *pmhpneses* is set to the default value for an SONET tributary
- *pmhpfeses* is set to the default value for an SONET tributary.

The attribute tcamd does not exist for SONET ports. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

Switching of interface standard – notification behavior for port attributes

When changing the interface standard *istd* using the ED-STMn command all resulting parameter changes are reported by ED-OCn notifications of the related port rate. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard – notification behavior for tributary attributes

When changing the interface standard *istd* using the ED-STMn command all resulting tributary parameter changes are reported by ED-STS1 notifications.

For efficiency reasons the ED-STS1 notifications are sent out as a single notification using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to

- 1 – 3, in case of an STM1/OC3 port
- 1 – 12, in case of an STM4/OC12 port
- 1 – 48, in case of an STM16/OC48 port
- 1 – 192, in case of an STM64/OC192 port
- 1 – 768, in case of an STM256/OC768 port.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard *istd* to SONET includes an change of *tcamd* from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard – PM impact

The change of interface standard has the following impacts w.r.t. existing PM registers:

1. all current bins related to the port will be reset
2. all history bins related to the port remain unchanged
3. all current bins related to tributaries being a member of the port will be reset
4. all history bins related to tributaries being a member of the port remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Generic ONNS specific behavior - Impact on *portControl* when changing *portClass*

When *portClass* is changed to TRADITIONAL, *portControl* is set to NORMAL, if not set differently within the same command.

Such a change is reported by an ED-OC48 notification for the parameter *portControl*.

Generic ONNS specific behavior - Impact on port parameter settings while *portClass* is TRADITIONAL

When *portClass* has the value TRADITIONAL for a port, setting of the following parameters are accepted, but shall have no impact until *portClass* is changed away from TRADITIONAL: *portControl*.

At the moment the *portClass* is changed away from TRADITIONAL the current values of these parameters are considered as valid.

portControl change with wildcard AID on port level

If the parameter *portControl* is changed and the value all is specified in the AID at port level, then the system shall change the parameter value for all ports that match the rate specified in the command *modifier*.

A mismatch of the port rate with the command *modifier* value shall not lead to error responses.

The system shall send DBCHG notifications for all ports, where the parameter *portControl* has been changed.

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *portClass* is changed to a value where this parameter is supported.

At the moment the *portClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

ONNS INNI Port related behavior - Impact on tributary mode setting when portClass is set to INNI

When *portClass* is set to INNI for a port, the parameter *trbmd* is set to “ADAPTIVE”. This are reported by an ED-OC48 notification for the parameter *trbmd*.

ONNS INNI Port related behavior - Impact on tributary parameter settings when *portClass* is set to INNI

When *portClass* is set to INNI, the parameter *hptimmon* is set to DISABLE for all tributaries located on this port. This are reported by ED-STS1 notifications for the parameter *hptimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to AUTO

When *portClass* is set to INNI and *ndetmd* is set to AUTO for a port, the parameter *pmode* is set to MONITORING. This are reported by an ED-OC48 notification for the parameter *pmode*.

ONNS INNI Port related behavior - Impact on port parameter settings when *portClass* is set to INNI and *ndetmd* is set to MAN

When *portClass* is set to INNI and *ndetmd* is set to MAN for a port, the parameters *rstimca* and *rstimmon* is set to DISABLE. This are reported by an ED-OC48 notification for the parameters *rstimca* and *rstimmon*.

ONNS INNI Port related behavior - Impact on port parameter settings while *portClass* is INNI

When *portClass* has the value INNI for a port, setting of the following parameters are inhibited:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

At the moment *portClass* is changed away from INNI the parameters are set back to their default values. This are reported by means of an ED-OC48 notification.

ONNS INNI Port related behavior - Impact on Neighbor related parameters when *portClass* is set to INNI

When *portClass* is set to INNI, each of the following parameters shall take its default value, unless a value is specified together with the transition of *portClass* to INNI:

- *ndetmd*
- *neighborASTNNodeID*

- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on neighbor related parameters if *portClass* is changed away from INNI

When *portClass* is changed away from INNI, the following parameters are set to their default values:

- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*

ONNS INNI Port related behavior - Impact on port parameter settings when *ndetmd* is set to AUTO

When *ndetmd* is set to AUTO for a port, the parameters *neighborNodeID* and *neighborPortID* shall keep their current values until automatic neighbor detection delivers new valid values.

When *ndetmd* is set to AUTO for a port, the parameter *neighborASTNNodeID* shall keep its current value until automatic neighbor detection delivers a new valid value.

ONNS EDGE Port related behavior - Impact on *loca*, *locz* when setting *portClass* from EDGE to TRADITIONAL when port is involved in a 2F or 4F BLSR/MS-SPRing protection group

For all tributaries of that port the *loca*, *locz* parameters are reset to their default values (empty string). This is reported by an ED-STS1 (*istd* = SONET for the port) or ED-VC3 (*istd* = SDH for the port) notification using a range over the port.

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, when *portClass* is set to EDGE

When the *portClass* is changed to EDGE, the following parameters of the EDGE parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to EDGE:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS EDGE Port related behavior - Impact on EDGE related port parameter settings, if *portClass* is changed away from EDGE

When *portClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

ONNS UNI PORT related behavior - Impact on tributary mode (*trbmd*) setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *trbmd* is set to FIXED. This are reported by an ED-rr notification for the parameter *trbmd*.

ONNS UNI PORT related behavior - Impact on *portControl* setting, when *portClass* is set to UNI

When *portClass* is set to UNI for a port, the parameter *portControl* is set to NORMAL. This are reported by an ED-rr notification for the parameter *portControl*.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, when *portClass* is set to UNI

When the *portClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *portClass* to UNI:

- *networkTNAType*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddress*
- *clientNodeID*
- *localLPID*

This is covered by corresponding validation rules.

ONNS UNI PORT related behavior - Impact on UNI related port parameter settings, if *portClass* is changed away from UNI

When *portClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

ONNS ENNI PORT related behavior - Impact on *ndetmd* when *portClass* is set to ENNI

When *portClass* is set to ENNI, the parameter *ndetmd* is set to MAN.

ONNS ENNI PORT related behavior - Impact on remaining ENNI related port parameter settings, when *portClass* is set to ENNI

When the *portClass* is changed to ENNI, the following parameters of the ENNI parameter set shall keep their actual values, unless specified differently in the command setting the *portClass* to ENNI:

- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *remTID*

For remaining ENNI parameters, values different to their default values are mandatory. This is covered by a corresponding validation rule.

ONNS ENNI PORT related behavior - Impact on ENNI related port parameter settings, if *portClass* is changed away from ENNI

When *portClass* is changed away from ENNI, the following parameters are set to their default values:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Example input/output

```
ED-STM64:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=644,UNEQUIPOSIG=644;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-STM16:1-1-#-#-1-1::inputsig=644,unequiposig=644:”
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the ED-STM64 command.

Validation rule: Enough bandwidth reserved for 2F protection group

Check of the tributary rate (inputsig/unequiposig) if the port is member of a 2f protection group. Then the assigned bandwith is max. half of the port's bandwith.

Error code:
`IDNV – Input, Data Not Valid, Assigned bandwidth exceeds half port's bandwith`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *inputsig* consistent with existing cross-connections

Check of the consistency of the structure of the input signal (*inputsig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: *unequiposig* consistent with existing cross-connections

Check of the consistency of the structure of the output signal (*unequiposig*) with existing cross-connections.

Error code:
`SNVS – Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port`

Validation rule: no cross-connections for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connections are still on this port.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: No lower order cross-connections, conversion cross-connections, loopbacks or substructure for *istd* change

Check that the interface standard parameter (*istd*) is not changed when lower order cross-connections, conversion cross-connections, lower order loopbacks or lower order substructure are still on this port. If lower order cross-connections, conversion cross-connections, loopbacks or substructure exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: no loopbacks for *istd* change

Check that the interface standard parameter (*istd*) is not changed when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no protection for *istd* change

Check that the interface standard parameter (*istd*) is not changed when this port is in an protection group.

Error code:

`SNVS – Status, Not in Valid State, a protection group is existing`

Validation rule: no test access session for *istd* change

Check that *istd* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: *servcond* must be enabled when timing source

Check that the *servcond* is not disabled when the port is used as a timing source.

Error code:

`SNVS – Status, Not in Valid State, Timing reference is assigned to this port`

Validation rule: *servcond* must be enabled when DCC enabled

Check that the *servcond* is not disabled when DCC channels are still existing for this port.

Error code:

`SNVS – Status, Not in Valid State, DCC channels are active for this port`

Validation rule: *servcond* must be enabled for protection ports

Check that the *servcond* is not OUTOFSERVICE if the port is used as protection port.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for protection ports`

Validation rule: *servcond* must be disabled for facility loopback

Check that the *servcond* is not enabled when a facility loopback is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Supply of outgoing J0 trace

Check that an outgoing J0 trace value (*strcout*) is supplied, too, if the format of the outgoing J0 trace (*strcwfmt*) is changed.

Error code:

`IDNC – Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`

Validation rule: transmitted path trace length

Check that the length of the outgoing J0 trace value (*strcout*) fits to the length given by outgoing J0 trace mode, i.e. write format (*strcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode.`

Validation rule: Change of FEC parameters only if supported by the current circuit pack

When changing one of the FEC-related parameters *fec*, *feccor*, and *fectype*, it is checked, whether the currently provisioned circuit pack supports this. If it is not supported the following error code is used:

Error code: `IDNV – Input, Data Not Valid, Parameter not supported`.

Validation rule: Enabling of Optical Channel only if supported by the current circuit pack

When enabling the optical channel via parameter *och* it is checked, whether the currently provisioned circuit pack supports this. If it is not supported the following error code is used:

Error code: `IDNV – Input, Data Not Valid, Parameter not supported`.

Validation rule: Enabling of FEC only while Optical Channel is enabled

The command is denied if FEC is enabled (*fec* = ENABLE) while the optical channel is disabled or if the optical channel is disabled (*och* = DISABLE) while FEC is enabled.

It is allowed to enable or disable FEC and the optical channel together in one command.

Error code:
`SNVS – Status, Not in Valid State, Optical Channel must be enabled`.

Validation rule: ’ALL’ qualifier used for wrong parameters

The ’ALL’ qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:
`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Check CCB against ptype and istd

Check wether the CCB fits to ptype and SDH standard.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=OC3 is made on a EP155 port which is provisioned as ptype=STM1E.

ONNS specific error codes (validation rules)

Validation rule: *portClass* only settable when *xcsetup* = SWISS

Modification of the *portClass* parameter is only allowed when xcsetup=SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule: No ONNS for interface standard change

Check that the interface standard parameter (*istd*) is not changed if ONNS functionality is available. This means:

- *portClass* must be TRADITIONAL or
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: *localLPID* already assigned to VCG port at VCAT layer

It is only allowed to set the parameter *localLPID* to a value different from its default value, if this value is not already assigned to the parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:
`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule: No setting of *minNNXCRate* to values larger than value of *maxNNXCRate* and vice versa

It is not allowed to configure parameters *maxNNXCRate* and *minNNXCRate* in a way, that the value of *minNNXCRate* is larger than the value of *maxNNXCRate*.

Error code:
`IDNV – Input, Data Not Valid, Min. and max. NN XC rates incompatible`

ONNS INNI port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI only possible, if service condition not OUTOFSERVICE

It is not allowed to change *portClass* to INNI, if servcond=OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Setting of *portClass* to INNI only possible, if no cross-connection existing

It is not allowed to change *portClass* to INNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Setting of *portClass* to INNI only possible, if no test access session active

It is not allowed to change portClass of a port to INNI when a tributary related to :this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to INNI

It is not possible to change *portClass* to INNI, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Setting of *portClass* to INNI not possible for ports in Protection Groups

It is not allowed to change portClass of a port to INNI, if this port is member of a Line/MS protection scheme.

Error code:

`SNVS – Status, Not in Valid State, Port must not be member of a protection group`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the portClass away from INNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from INNI

It is not possible to change the *portClass* away from INNI, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled:

- No cross-connection may exist on any tributary of the port
- *portControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Setting of *portClass* away from INNI only possible, if the port is not in use by M:N reservation

For setting *portClass* from INNI to TRADITIONAL or EDGE, the following must be fulfilled: No reservation related to a protecting NN connection within an M:N protected connection group may exist on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Changing of *portClass* from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port

For changing the value of the parameter *portClass*, no DCC of the port must be enabled for the SCN.

Error code:

`SNVS – Status, Not in Valid State, DCC on SCN may not be enabled`

Validation rule: *servcond* must be enabled when portClass = INNI

Check that the servcond is not OUTOFSERVICE when *portClass* = INNI.

Error code:

`SNVS – Status, Not in Valid State, can not disable servcond for INNI ports`

Validation rule: Some parameters not settable for ports of *portClass* = INNI

The following parameter is not changeable to value FIXED for ports with *portClass* = INNI:

- *trbmd*.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: Port Mode not settable for ports of *portClass* = INNI

The system shall not allow to change the parameter *pmode* if *portClass* has been set to INNI prior to this command or if *portClass* is set to INNI with this command. The parameter *pmode* shall either be omitted or specified with the only allowed value MON.

Error code:

`SNVS – Status, Not in Valid State, port mode change not allowed for INNI ports`

Validation rule: Some Parameters not settable for ports of *portClass* = INNI

The following parameters is not changeable for ports with *portClass* = INNI:

- *exprstrc*
- *rstimca*
- *rstimmon*
- *strcfmt*
- *strcout*
- *strcwfmt*

Error code:

`SNVS – Status, Not in Valid State, J0 parameters can not be set for INNI ports`

Validation rule: Setting of INNI properties only possible if *portControl* is LOCKOUT

The system shall allow to change the following parameters only under the conditions:

1. if the port control is LOCKOUT while port class is INNI
2. together with the transition of port control from LOCKOUT to NORMAL while port class is INNI or
3. together with the transition of port class to INNI

The parameters are:

- *adminCost*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborASTNNodeID*
- *neighborNodeID*
- *neighborPortID*
- *ndetmd*
- *propagationDelay*
- *srlgList*

Error code:

`SNVS – Status, Not in Valid State, only allowed for INNI port in lockout state`

Validation rule: Check of INNI parameters

It is not allowed to change one of the following parameters on a port with portClass = INNI, if NN cross-connections exist on that port:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for change of iNNI parameters`

Validation rule: Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing

It is not allowed to change one of the following parameters on a port with *portClass* = INNI, if reservations w.r.t. M:N protecting NN connections on that port are existing:

- *neighborNodeID*
- *neighborPortID*
- *propagationDelay*
- *srlgList*
- *neighborASTNNodeID*
- *maxNNXCRate*
- *minNNXCRate*

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters`

Validation rule: Removing of entries from *srlgList* not allowed if NN cross-connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while NN cross-connections exist at the port.

Error code:

`SNVS – Status, Not in Valid State, no XCs allowed for removing of srlgList entries`

Validation rule: Removing of entries from *srlgList* not allowed if reservations wrt. M:N protecting connections exist

The system shall not allow to remove existing entries from the Shared Risk Link Group Identities List (*srlgList*) at a port with *portClass* INNI while reservations wrt. M:N protecting NN connections exist on that port.

Error code:

`SNVS – Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries`

Validation rule: Some parameters not settable for ports of *ndetmd* AUTO

The following parameters is not changeable for ports with ndetmd AUTO:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `SNVS – Status, Not in Valid State,` *`ndetmd`* `must not be AUTO`

Validation rule: Additional parameters required when setting *ndetmd* to MAN

Values for the following parameters must be given, if *ndetmd* is changed to MAN:

- *neighborNodeID*
- *neighborPortID*
- *neighborASTNNodeID*

Error code: `IDNV – Input, Data Not Valid, additional parameters required`

Validation rule: No setting of *neighborASTNNodeID* or *neighborNodeID* to corresponding IDs of local node

It is not allowed to configure parameters *neighborASTNNodeID* or *neighborNodeID* to corresponding values of the local node itself, i.e. it is not possible to configure a node to be its own ONNS neighbor.

Error code:

`IDNV – Input, Data Not Valid, node itself is not a valid neighbor`

Validation rule: No setting of duplicate *neighborASTNNodeID* and *neighborPortID* parameter combination

Setting of the parameters *neighborASTNNodeID* and *neighborPortID* is only possible, if the same combination of values does not already exist for a different INNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, duplicate neighbor port information`

Validation rule: Setting of neighbor information to UNKNOWN

In order to set neighbor information to UNKNOWN, zero values must be set for all parameters *neighborASTNNodeID*, *neighborNodeID* and *neighborPortID* consistently. It is not possible to set zero value for only one or two of them.

Error code:

`IDNV – Input, Data Not Valid, zero values can only be set for all neighbor information consistently`

ONNS EDGE port related error codes (validation rules)

Validation rule: Setting of *portClass* to INNI or EDGE only possible, if no test access session active

It is not allowed to change *portClass* of a port to INNI or EDGE when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change *portClass* to EDGE, if a cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to EDGE

It is not possible to change the *portClass* to EDGE, if there exist 1:N Broadcasts with N>2.

Error code:

`SNVS – Status, Not in Valid State, Broadcast with N greater 2 exists`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1+1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Setting of *portClass* from TRADITIONAL to EDGE

For setting *portClass* from TRADITIONAL to EDGE, the port must not be the protection port of a 1:1 MSP group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from EDGE

It is not possible to change the *portClass* away from EDGE, if the target of *portClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Setting of *portClass* from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:

For setting *portClass* from EDGE to TRADITIONAL the following must be fulfilled:

- No cross-connections of xconcls = NN may exist on any tributary of the port.
- *portControl* must be LOCKOUT.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of EDGE parameters

It is possible to modify the EDGE related parameters only

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to EDGE

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule: Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports

When assigning TNA information to an EDGE port, a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Consistent Parameters setting required when removing TNA information from EDGE ports

When removing TNA information from an EDGE port, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule: Modify of EDGE parameters

It is not possible to modify any of the EDGE related parameters, if there still exist any cross-connections for the EDGE port, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* is already assigned to any UNI port in the system.

Error code:
`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

This validation means, that UNI and EDGE ports is not mixed under one TNA.

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set non-default TNA parameters for an EDGE port, if the desired value of *networkTNAAddress* already assigned as *networkTNAAddrVCAT* to any VCG in the system.

Error code:
`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

Validation rule: Setting of TNA information only if ENNI Node exists

It is not possible to modify any of the EDGE related parameters to values different from their default values, if the ENNI node does not exist in the system.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code: `SNVS – Status, Not in Valid State, ENNI node must exist`

ONNS UNI port related error codes (validation rules)

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if cross-connections on that port are still existing.

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:
`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1+1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1+1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change the *portClass* to UNI, if the port is the protection port of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a 1:1 protection port`

Validation rule: Change the *portClass* to UNI

It is not possible to change *portClass* to UNI, if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameter

It is not possible to change the *portClass* to UNI, unless valid values for the following UNI parameters are given:

- *clientNodeI*
- *networkTNATyp*
- *networkTNAAddres*
- *localLPID*

Error code:

`IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT value`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from UNI

It is not possible to change the *portClass* away from UNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the UNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to UNINode`

Validation rule: Check of parameters for UNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for UNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of *portClass* UNI

The following parameter is not changeable to the value ADAPTIVE for ports with *portClass* UNI:

trbmd

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: Modify of UNI parameters

It is possible to modify the UNI related parameters only:

1. if the *portControl* state is LOCKOUT
2. during the transition of the *portClass* to UNI.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for UNI port in lockout state`

Validation rule: Modify of UNI parameters

It is not possible to modify any of the UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*
- *clientNodeID*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, when *portClass* is equal to UNI or during setting *portClass* to UNI:

- *clientNodeID*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Network TNA Address Uniqueness per UNI client

It is not possible to set TNA parameters for a UNI port, if the desired value of *networkTNAAddress* is already assigned to another UNI port in the system, but different values of *clientNodeIDs* are associated to both ports.

Error code:

`IDNV – Input, Data Not Valid, TNA Address not allowed for the Client Node ID`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned to any EDGE port in the system.

Error code:

`IDNV – Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA`

Validation rule: TNA parameters already assigned on VCAT layer

It is not possible to set TNA parameters for a UNI port if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddrVCAT* to any VCG port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA already assigned to a VCG port`

ONNS ENNI PORT related error codes (validation rules)

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if *servcond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, port is out of service`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Setting of *portClass* to ENNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist

Check if LO cross-connections, substructure or conversion cross-connections exist which involve the port to be changed. If LO cross-connections, substructure or conversion cross-connections exist, the following error message is returned:

Error code:

`SNVS – SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections are still existing`

Validation rule: Change the *portClass* to ENNI

It is not possible to change *portClass* to ENNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if cross connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, cross connect loopback exists`

Validation rule: Change the *portClass* to ENNI

It is not possible to change the *portClass* to ENNI, if the port is any member of a 1:1 MSP/APS group.

Error code:

`SNVS – Status, Not in Valid State, Port must not be a member of a 1:1 protection group`

Validation rule: Check of ENNI parameters

It is not possible to change the *portClass* to ENNI, unless valid values for the following ENNI parameters are given:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, Valid values for ENNI parameters required`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if *portControl* is NORMAL. First *portControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the target of *portClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if there still exist any NN cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Change the *portClass* away from ENNI

It is not possible to change the *portClass* away from ENNI, if the DCC-R (Section DCC) or DCC-M (Line DCC) on this port is assigned to the ENNINode.

Error code:
`SNVS – Status, Not in Valid State, DCC must not be assigned to ENNINode`

Validation rule: Check of parameters for ENNI

It is not possible to modify the parameter *servcond* to OUTOFSERVICE for ENNI ports.

Error code:
`SNVS – Status, Not in Valid State, can not change servcond for this port class`

Validation rule: Some parameter values not settable for ports of portClass ENNI

The following parameter is not changeable to value AUTO for ports with *portClass* ENNI:

- *ndetmd*

Error code: `SNVS – Status, Not in Valid State, portClass must not be ENNI`

Validation rule: Modify of ENNI parameters

It is possible to modify the ENNI related parameters only:

- if the *portControl* state is LOCKOUT
- during the transition of the *portClass* to ENNI.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *remRAID*
- *adminCost*
- *srlgList*
- *linkResCls*
- *maxNNXCRate*

- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*
- *remTID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for ENNI port in lockout state`

Validation rule: Modify of ENNI parameters

It is not possible to modify any of the ENNI related parameters, if there still exist any cross-connection for the ENNI port.

The parameters are:

- *remAdvNodeID*
- *localLinkID*
- *remLinkID*
- *linkResCls*
- *maxNNXCRate*
- *minNNXCRate*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Unique setting of *localLinkID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is unique within the system.

Error code:

`IDNV – Input, Data Not Valid, local Link ID value must be unique`

Validation rule: Unique setting of *localLinkID* wrt. *localLPID*

It is only allowed to set the parameter *localLinkID* to a value different from its default value, if this value is not already assigned to parameter *localLPID* of any port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule: Check of ENNI parameters

It is not possible to set the following ENNI parameters to their default values if *portClass* is equal to ENNI or during the setting of *portClass* to ENNI:

- *remAdvNodeID* (if a corresponding level 1 routing model is defined)
- *localLinkID*
- *remLinkID*
- *remRAID*
- *neighborSigPCID*
- *neighborSigPCIPAddress*

Error code:

`IDNV – Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`

Related TL1 messages

"RTRV-STM64: Retrieve STM64 port" (p. 3-2926)

References

GR-199-CORE (ED-*rr*)

ED-STM64 autonomous notifications

LOS for transmission port STM64 clears

If LOS is cleared and current port mode is AUTO, then *portmode* is updated (new value: MON). This update is reported via ED-STM64 notification with parameter pmode.

Change of neighbor node and neighbor port information

If *ndetmd* = AUTO a change of the attributes *neighborNodeID*, *neighborPortID* or *neighborASTNNodeID* is notified by the related ED-STM64 notification.

ED-STM64 notification

Output format

Syntax: **`ED-STM64`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: SPortAID of ST-64 port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = BURST.
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = BURST.
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps.
   This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC-64C.
6. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = POISSON.
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter is allowed when the related STM64 port is created, it is used when *sdfsmode* = POISSON.
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.

Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`fec`** – FEC Enable
Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
Type: Alphanumeric value set.
Range:
DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768 / STM256 basis low cost intra office 1310 nm optics.

   ENABLE / DISABLE for:
   - OC192 / STM64 OLS800G compatible 80 colour optics
   - OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
   - OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

   ENABLE (only) for:
   - OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G.

   Default:
   DISABLE for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768 / STM256 basis low cost intra office 1310 nm optics.

   ENABLE for:
   - OC192 / STM64 OLS800G compatible 80 colour optics
   - OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
   - OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
   - OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

10. **`femspm`** – Far-end Multiplex Section Enable
    Enable or disable of the performance monitoring of the multiplex section.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
11. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: (set a forced DUS/DNU)
    - DISABLE (default): (clear a forced DUS/DNU).
12. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
13. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port’s bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
    Default: 644.
14. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range:
    - SONET
    - SDH.

    Default: As specified via *istddflt* on system level, see definition in “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400).

15. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
16. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
17. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

18. **maxNNXCRate** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3, VC4, VC44C, VC416C or VC464C (default)
19. **minNNXCRate** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3 (default), VC4, VC44C, VC416C or VC464C.
20. **msrei** – MS-REI mode
    Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
    Type: Alphanumeric value set.

    Range:

    - SINGLE
    - EXTENDED (default).
21. **ndetmd** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN

22. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.
23. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the ONNS application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.
24. **`neighborPortID`** – Neighbor Port Id
    This identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is introduced specifically for use by NN application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

25. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

27. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

28. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

29. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

30. **`och`** – optical channel enable
    Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don't have any impact on transmission.
    Type: Alphanumeric value set.
    Range:
    - for ports on OP10 packs, which support to enable or disable OCH:: ENABLE: enable optical channel; DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
    - for client ports on packs which support OTN: DISABLE: disable, optical channel not applicable.

    Default:
    - for OP10 and OP40 packs, which cannot support OCH: DISABLE
    - for OP10 packs, which cannot support OCH: ENABLE
    - for OP10 and OP40 packs, which only support OCH: ENABLE
    - for client ports on packs which support OTN: DISABLE.
31. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

32. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

33. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

    Default:

    - AUTO: for optical SDH/SONET ports
    - NMON: for client ports-

34. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000

Default: The default values differ for SDH and SONET and the different rates:

- SDH (all rates): 2400
- SONET OC48: 2392
- SONET OC192: 8554
- SONET OC768: 22776.

35. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.

36. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

37. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

38. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

39. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

40. **remLinkID** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

41. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

42. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

43. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
44. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
45. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
46. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
47. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC192: Retrieve OC192 port" (p. 3-2664)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

48. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.
49. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
50. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.
51. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
    If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

52. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string dependent on the strcwfmt parameter.

    Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
53. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
54. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

55. **`trbmd`** – Tributary mode
If trbmd is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If trbmd is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED
Default: The initial value is defined as the trbmddflt parameter on system level, see definition in RTRV-EQPT (system) command.

56. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.

Example output

```
ED-STM64:UNITE-NE:1-1-#-#-1-1:123456:::INPUTSIG=644,UNEQUIPOSIG=644;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-STM16:1-1-#-#-1-1::inputsig=644,unequiposig=644:″
;
```

ED-STS1: Edit STS1 tributary

The command **ED-STS1** is used to provision all parameters related to the tributary STS1.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-STS1**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   AID of an SONET tributary. The keyword **all** is allowed on slot, port and tributary level. The so called ″dynamic parameters″ is reported, if a **all** is used at the tributary level.

   The ″dynamic parameters″ are:

   - *ptrc*, *ptrcst*, *siglb*, *trailstate*.

   Type: AID
   Range: STS1NumAID.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   If the STS-1 tributary is related to a EC1 port, then the parameter *sdsfmode* is fixed to POISSON. As a consequence, the parameter *brstintvl* is settable but will never be used by the system in this case.
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   If the STS-1 tributary is related to a EC1 port, then the parameter *sdsfmode* is fixed to POISSON. As a consequence, the parameter *brstthr* is settable but will never be used by the system in this case.
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
4. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
   Limitation of STS1 related to EC1 ports: For STS1 tributaries related to EC1 ports, the only useful *dexcthr* value is -5. Only this value will lead to Excessive Bit Error alarms. The other values are kept for compatibility reasons.
5. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
6. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

7. **hptimmon** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **loca** – Ring A-node TID
   This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
   loca is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in “RTRV-HDR: Retrieve header” (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
9. **locz** – Ring Z-node TID
   This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
   locz is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
10. **neppm** – Near-End Path PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
11. **pjepm** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

12. **`pmhpfeses`** – Performance Monitoring Path Far-End SES Threshold
This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
13. **`pmhpneses`** – Performance Monitoring Path Near-End SES Threshold
This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
14. **`pse`** – PDI-P Switching Enable
The PDI-P Switching Enable parameter indicates if the PDI-P contributes to the automatic path protection switch criteria.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
15. **`ptca`** – Path TCA Profile Pointer
Type: Alphanumeric string.
Range: string with up to 24 characters.
Default: DEFAULT0.
16. **`ptrcrfmt`** – Path Trace Read Format (J1)
The read format of J1, the incoming path trace (ptrc).
Type: numeric string.
Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string.
    - 64: for 64 byte string.
    Default: 2.
17. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in BURST or POISSON mode.
Type: Alphanumeric value set.

Range:

- BURST
- POISSON (default).

18. **`tmonmd`** – Tributary Monitoring Mode
    This parameter indicates the mode of the tributary.
    Type: Alphanumeric value set.
    Range:
    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.
19. **`tribsonetpn`** – Tributary Alarm Severity Assignment Profile
    This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.

Output format

If the ED-STS1 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command **`ED-STS1`** is also applicable to STS3c and other concatenated tributaries.

Even if the command is used for a concatenated signal, it can be applied to any of the tributaries (for example, to the second tributary of the concatenated signal). However, only the parameters pertaining to the first tributary of a concatenated signal are used by the software to determine system operation. If the signal were changed from a concatenated signal to non-concatenated, then the parameters for each of the tributaries would be used by the software.

The command allows to modify all attributes listed above as input spec-block parameters. The modification is reported by means of an ED-STS1 notification for tributary.

Change of J1 expected format - FM impact

When the expected J1 format (ptrcrfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J1 (exphptrc) and the currently received J1 (ptrc).

When the expected J1 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Signal degrade/signal fail mode provisioning

Refer to the detailed behavior of the "ED-VC3: Edit VC3 tributary" (p. 3-1339) command for details on Signal Degrade/Signal Fail Mode Provisioning for tributaries.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term "PM monitor" has the same semantics as the term "PM package". The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Disabling of PJE Monitor

The disabling of a pointer justification monitor shall have the following impacts w.r.t. existing PJE registers:

1. stop PJE counting
2. all current bins related to the PJE monitor is deleted
3. all history bins related to the PJE monitor is deleted

Example input/output

```
ED-STS1:UNITE-NE:1-1-#-#-3-1-1:123456:::tmonmd=MON,tribsonetpn=Default,degthr=-
      6,dexcthr=-3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30;

   UNITE-NE 00-10-14 14:59:09
M  123456 COMPLD
;
```

```
   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ”TIME=14-59-09,DATE=00-10-14:ED-STS1:1-1-#-#-3-1-1::
      tmonmd=MON,tribsonetpn=Default,degthr=-6,dexcthr=-3,ptrcrfmt=1,
ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-STS1 command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Not more than one PJE PM monitor per port

Check that no PJE PM monitor is already enabled for any tributary of the port to which the addressed tributary belongs to.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PJE counters exceeded`

Validation rule: Check CCB against *istd* of related port

Check wether the CCB fits to istd of the related port

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

Validation rule (ONNS specific): *loca*, *locz* changes only allowed when *portClass* = EDGE and part of a BLSR/MS-SPRING group

Setting the *loca*, *locz* parameters away from the default value is only allowed when the related *portClass* = EDGE and the port is part of a BLSR/MS-SPRING protection group.

Error code:

`IDNV – Input, Data Not Valid, related portClass not EDGE or port not in BLSR/MS-SPRING group`

Validation rule (ONNS specific): *loca*, *locz* changes only allowed when tributary not involved in a cross-connection

Setting the *loca*, *locz* parameters are inhibited when the tributary is already input or output tributary of a cross-connection.

Error code:

`IDNV – Input, Data Not Valid, tributary already part of a cross-connection`

Validation rule (ONNS specific): Maximum number of tributaries with *loca*, *locz* entries must not exceed 100

It is checked that the maximum number of triburaries with loca, locz parameters set (i.e. having a value different from the default value) does not exceed 100.

Error code:

`SSRE – Status, System Resources Exceeded, limit of tribs with loca, locz exceeded`

Validation rule (ONNS specific): *hptimmon* changes to ENABLE only allowed when corresponding port is not of *portClass* INNI

Setting the *hptimmon* parameter to ENABLE is only allowed when the port the VC-3 is located on does not have the port class INNI.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Related TL1 messages

“RTRV-VC3: Retrieve VC3 tributary” (p. 3-3069)

References

GR-199-CORE (ED-*rr*)

ED-STS1 notification

Output format

Syntax: **ED-STS1**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   AID of an SONET tributary. The keyword **all** is allowed on slot, port and tributary level. The so called ″dynamic parameters″ is reported, if a **all** is used at the tributary level.
   The ″dynamic parameters″ are:
   - *ptrc*, *ptrcst*, *siglb*, *trailstate*.

   Type: AID
   Range: STS1NumAID.

Spec block parameters

1. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **brstthr** – Bursty Error Threshold
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **crslpbkrate** – Cross-connect loopback rate
   It is independent from the current interface standard and therefore both SDH and SONET values are allowed.
   The parameter *crslpbkrate* is not reported if the parameter *crlspbkstat* indicates that no loopback is set.
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C

- STS1
- STS3
- STS12
- STS48
- STS192

4. **`crslpbkstat`** – Cross-connect loopback status
   State of cross- connect loopback.
   Type: Alphanumeric value set.
   Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
5. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
6. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
7. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
8. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`hptimmon`** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **loca** – Ring A-node TID
    This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
    loca is introduced specifically for use related to ONNS application.
    Type: TID or empty-string.
    Range: Please refer to the TID definition in "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: "". This is displayed as \"\" in output direction.
11. **locz** – Ring Z-node TID
    This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
    locz is introduced specifically for use related to ONNS application.
    Type: TID or empty-string.
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: "". This is displayed as \"\" in output direction.
12. **neppm** – Near-End Path PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
13. **pjepm** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).

14. **`pmhpfeses`** – Performance Monitoring Path Far-End SES Threshold
This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
15. **`pmhpneses`** – Performance Monitoring Path Near-End SES Threshold
This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
16. **`pse`** – PDI-P Switching Enable
The PDI-P Switching Enable parameter indicates if the PDI-P contributes to the automatic path protection switch criteria.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
17. **`ptca`** – Path TCA Profile Pointer
Type: Alphanumeric string.
Range: string with up to 24 characters.
Default: DEFAULT0.
18. **`ptrcrfmt`** – Path Trace Read Format (J1)
The read format of J1, the incoming path trace (ptrc).
Type: numeric string.
Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string.
    - 64: for 64 byte string.
    Default: 2.
19. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in BURST or POISSON mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).

If the tributary is related to a EC1 port, then the range is limited to: POISSON

20. **`substr`** – Substructured
    This parameter indicates if the tributary is substructured.
    Substructure for LO cross-connections:
    A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.
    The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).
    A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **`ED-SUBS`** command.
    Substructure for conversion cross-connections:
    A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.
    The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.
    A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.
    Type: Alphanumeric value set.
    Range:
    - YES: Automatically sub structured for LO cross-connections
    - MANSUB: Manually sub structured for LO cross-connections
    - CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
    - NO (default): Not sub structured, not used for conversion cross-connections

    If the STS-1 tributary is related to a EC1 port, then no LO cross-connections are allowed. As a consequence, the parameter *substr* can only have the values CNV and NO in this case.
21. **`tmonmd`** – Tributary Monitoring Mode
    This parameter indicates the mode of the tributary.
    Type: Alphanumeric value set.

Range:

- MON = alarm monitoring is taking place on the port.
- NMON (default) = alarm monitoring is not occurring on the port.

22. **`tribsonetpn`** – Tributary Alarm Severity Assignment Profile
This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

Example output

```
ED-STS1:UNITE-NE:1-1-#-#-3-1-1:123456:::tmonmd=MON,tribsonetpn=Default,degthr=-
      6,dexcthr=-3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30;

   UNITE-NE 00-10-14 14:59:09
M  123456 COMPLD
;

   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ″TIME=14-59-09,DATE=00-10-14:ED-STS1:1-1-#-#-3-1-1::
      tmonmd=MON,tribsonetpn=Default,degthr=-6,dexcthr=-3,ptrcrfmt=1,
ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30:″
;
```

ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary.

The **ED-SUBS** command:

- creates the lower order substructure and lower order tributaries at an STS1or VC4 tributary as preparation of the lower order cross-connection creation.
- deletes the lower order substructure and lower order tributaries to make the STS1or VC4 tributary usable for higher order cross-connections.

Note: This command is not applicable for the TransMUX card, this is restricted by the AID type.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-SUBS**-*modifier:tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the high-order tributary which is going to be substructured.
   Type: Alphanumeric value set.
   Range: STS1, VC4
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   AID of an SDH or SONET tributary located on a I/O pack.
   Type: AID.
   Range: STS1NumAID
4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec bloc parameters

1. **`substr`** – Substructured
   The parameter substr specifies whether the tributary is substructured or de-substructured. The parameter *substr* is mandatory. A substructured HO tributary has a HO path termination associated which provides LO tributaries to be used for LO cross-connections. A substructured HO tributary cannot be used to set up HO cross-connections.
   Type: Alphanumeric value set.

   Range:

   - MANSUB to create the LO substructure
   - NO to delete the LO substructure.

Output format

If the **`ED-SUBS`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Detailed behavior description

Update of HO tributary

The system shall update the parameter substr of the specified HO tributary to the value MANSUB. The parameter value change is notified via ED-STS1 or ED-VC3 Notification depending on the interface standard of the HO tributary.

Creation of HO Path Termination

If modifier has the value STS1, the system shall create one HO STS1 Path Termination Function with the default LO tributary structure of 28 VT1.5, i.e. the parameters inputsig and unequiposig set to the value 2811.

If modifier has the value VC4, the system shall create one HO VC4 Path Termination Function with the default LO tributary structure of 63 VC12, i.e. the parameters inputsig and unequiposig set to the value 6312.

The creation of the HO Path termination function is notified via ENT-PTF Notification.

Creation of LO Tributaries

If modifier has the value STS1, the system shall create 28 LO VT1.5 tributaries.

If modifier has the value VC4, the system shall create 63 LO VC12 tributaries.

The creation of the LO tributaries are notified via one ENT-LONIM Group Notification, i.e. using the AID ranging.

Deletion of LO Tributaries

The system shall delete all LO tributaries related to the HO tributary specified in the command.

The deletion of the LO tributaries are notified via one DLT-LONIM Group Notification, i.e. using the AID ranging.

Deletion of HO Path Termination

The system shall delete the HO Path Termination Function related to the HO tributary specified in the command.

The deletion of the HO Path Termination Function is notified via one DLT-PTF Notification.

Update of HO tributary

The system shall update the parameter substr of the specified HO tributary to the value NO. The parameter value change is notified via ED-STS1 or ED-VC3 Notification depending on the interface standard of the HO tributary.

Error responses

Note: In a system with provisioned LOXC40 pack, the following LO substructuring related TL1 commands are denied with wrong error codes:

- ED-SUBS command on an already substructured HO tributary is denied with wrong error code: SNVS – Status Not in Valid State, LOXC switch capacity Invalid
  `SNVS – Status Not in Valid State, LOXC switch capacity Invalid`
  The correct error code should be:
  `SNVS – Status Not in Valid State, HO tributary already substructured`
- ED-SUBS command on a HO tributary at a BLSR port is denied with wrong error code:
  `IDNC-Input Data Not Consistent, No BLSR Interworking with low-order traffic`
  The correct error code should be:
  `IDNC-Input Data Not Consistent, No BLSR/MS-SPRing Interworking with low-order traffic`
- De-provisioning of the LOXC40 pack with DLT-EQPT command while a HO tributary is substructured, is denied with wrong error code:
  `SNVS-Status, Circuit pack is in use`
  The correct error code should be:
  `SNVS-Status, Not In Valid State, LO cross-connections or substructure still exist`

Validation rule: Check CCB against istd

Check wether the CCB fits to *istd* of the related port.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

Validation rule: Entities must exist

Command processing shall verify that the HO tributary specified in the command exists.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Command not allowed while conversion cross-connections exist

When the command is applied at a HO tributary, then the command processing shall verify that no AU3/AU4 conversion cross-connection exists at the HO tributary or at LO tributaries related to the HO tributary.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: LO Substructure creation not allowed on already substructured tributaries

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the high-order tributary specified in the command is not already substructured, i.e. that the parameter substr of the HO tributary has the value NO.

Error code:

`SNVS – Status, Not in Valid State, HO tributary already substructured`

Validation rule: Substructure deletion not allowed with LO cross-connections existing

When an attempt is made to delete the substructure at a HO tributary, the command processing shall verify that no LO cross-connections exist at LO tributaries related to the HO tributary.

Error code:

`SNVS – Status, Not in Valid State, lower order cross-connections are still existing`

Validation rule: Substructure deletion not allowed with LO cross-connect loopbacks existing

When an attempt is made to delete the substructure at a HO tributary, the command processing shall verify that no LO cross-connect loopbacks exist at LO tributaries related to the HO tributary.

Error code:

`SNVS – Status, Not in Valid State, lower order cross-connect loopbacks are still existing`

Validation rule: Substructure deletion not allowed with LO test access sessions existing

When an attempt is made to delete the substructure at a HO tributary, the command processing shall verify that no LO cross-connect loopbacks exist at LO tributaries related to the HO tributary.

Error code:

`SNVS – Status, Not in Valid State, lower order test access sessions are still existing`

Validation rule: Entities must fit LOXC interface standard

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the specified modifier and the given aid's fit to the LOXC overall interface standard, i.e. to the value of the parameter lostd related to the active LOXC pack.

Error code:

`IDNV – Input, Data Not Valid, Low-order cross-connections of this interface standard not supported`

Validation rule: LO Substructure creation not allowed on already cross-connected tributaries

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the high-order tributary specified in the command is not used in a HO cross-connection.

Error code: `SNVS – Status, Not in Valid State, HO cross-connection exists`

Validation rule: LO Substructure creation not allowed while loopback exists

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that no HO cross-connection loopback exists at the high-order tributary specified in the command.

Error code:

`SNVS – Status, Not in Valid State, Tributary has existing loopback`

Validation rule: LO Substructure creation not allowed while test session exists

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that no HO test access session exists at the high-order tributary specified in the command.

Error code:

```
SNVS – Status, Not in Valid State, Tributary used in test access session
```

Validation rule: HO tributary rate consistency

Command processing shall verify that the rate of the HO tributary specified in the command fits to the modifier parameter.

Error code:

```
IDNC – Input, Data Not Consistent, Specified Tributary does not support
the indicated rate
```

Validation rule: LO Substructure creation not allowed on ports in adaptive mode

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the related port is in fixed mode.

Error code:

```
SNVS – Status, Not in Valid State, Tributary at port in adaptive mode
```

Validation rule: LO Substructure creation not allowed on BLSR/MS-SPRing ports

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the related port is not involved in a BLSR or MS-SPRing protection group.

Error code:

```
DNC – Input, Data Not Consistent, No BLSR/MS-SPRing inter-working with
low-order traffic
```

Validation rule: LOXC pack with more than 0G switch capacity

It is not possible to create the substructure at a HO tributary, if there is no LOXC pack with more than 0 GBit/s low-order switch capacity.

Error code:

```
SNVS – Status, Not in Valid State, LOXC switch capacity invalid
```

Validation rule: Pack restrictions for LO substructure creation

Command processing shall verify that the specified HO tributary does not reside on one of the following packs:

- the OPT2G5 pack,
- the EP51 pack (DS3 and EC1 interfaces),
- the GE1 pack,
- the GE10PL1 pack.

Error code:

`IDNC – Input, Data Not Consistent, LO tributary not supported by that pack`

Validation rule: LO Substructure creation not allowed on INNI ports

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the related port does not have the Port Class INNI.

Error code:

`IDNC – Input, Data Not Consistent, Tributaries must not be on ports of portClass INNI`

Validation rule: LO Substructure creation not allowed on UNI ports

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the related port does not have the Port Class UNI.

Error code:

`IDNC – Input, Data Not Consistent, Tributaries must not be on ports of portClass UNI`

Validation rule: LO Substructure creation not allowed on ENNI ports

When an attempt is made to create the substructure at a HO tributary, the command processing shall verify that the related port does not have the Port Class ENNI.

Error code:

`IDNC – Input, Data Not Consistent, Tributaries must not be on ports of portClass ENNI`

Related TL1 messages

- "RTRV-VC3: Retrieve VC3 tributary" (p. 3-3069)
- "RTRV-STS1: Retrieve STS1 tributary" (p. 3-2946)
- "RTRV-PTF: Retrieve tributary with path termination function" (p. 3-2800)
- "RTRV-LONIM: Low-order non-intrusive monitor" (p. 3-2486)
- "ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)
- "ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)

ED-SYNCDATA: Synchronization of ONNS data

Overview

The ED-SYNCDATA-mod command is used to have ONNS data to be synchronized based on the result of audit operation. AUD-mod command is assumed to precede this data-sync request. The modifier of the ED-SYNCDATA-mod command indicates the scope of data to be synchronized.

ED-SYNCDATA-LOCAL

ED-SYNCDATA-LOCAL performs synchronization on ONNS data with the master data from local host NE. They are cross-connect map and port objects. Data-sync shall always be preceded by data-audit operation. Changes to ONNS data are made only if there is any discrepancy found via data-audit.

For mismatched port object cases, the following updates are made to ONNS data:

- port object exists only in ONNS data, but not in host NE data the corresponding port object is removed from NN data.
- port object exists only in host NE data, but not in NN data the corresponding port object is added to NN data.

For mismatched cross-connect cases, the following updates are made to NN data:

- NN owned cross-connect exists only in NN data, but not in host NE the corresponding NN cross-connect and its associated path information are removed from NN data
- NN owned cross-connect exists only in host NE data, but not in NN data the corresponding NN cross-connect is added to NN data with a dummy pathID
- non-NN owned cross-connect exists only in NN data, but not in host NE the corresponding cross-connect is removed from NN data
- non-NN owned cross-connect exists only in host NE data, but not in NN data the corresponding cross-connect is added to NN data with non-NN ownership.

NN should respond to the request in two steps: first, the result of data-audit with all the mismatched cases, and then a indication of the successfulness of the data-sync operation. The *dummy pathID* is an ID assigned by local node as a result of local data-sync operation (see ED-SYNCDATA-LOCAL command for details). It should be in pathID format and unique.

ED-SYNCDATA-PATH

EDSyncData-PATH modifies an ONNS path ID associated with local cross-connect data with valid path ID. Note that CONN data repository is not affected by this operation.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No.

Input format

Command syntax of ED-SYNCDATA-LOCAL:
`ED-SYNCDATA-LOCAL:`*srcTid*:*srcAid*:*ctag*;

Command syntax of ED-SYNCDATA-PATH:
`ED-SYNCDATA-PATH:`*srcTid*:*srcAid*:*ctag*:*oldPthID*,*newPathID*;

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing “.”). It should be a unique value for each new command.

4. **`oldPathID`**
   Current Path Identifier from the local ONNS path data. Old path ID may be a dummy ID assigned as a result of local ONNS data synchronization.
   Type: character string of length up to 20.
   Range: nodeID-Index; nodeID (four dot separated numeric strings: each string 0..255) Index (Numeric string).
   Example: 135.13.196.203-43216
5. **`newPathID`**
   New Path Identifier which will replace the oldPathID.
   Type: character string of length up to 20.
   Range: nodeID-Index; nodeID (four dot separated numeric strings: each string 0..255) Index (Numeric string).
   Example: 135.13.196.203-43216

Output format

NN should respond to the request in two steps: first, if completed successfully, the result of data-audit with all the mismatched cases, and then a indication of the successfulness of the data-sync operation. The output format for the result of data-audit should be the same as the one for the response to **`AUD`** command.

```
   srcTid date time
M  ctag COMPLD
;
```

Whenever a modification is made to any ONNS path data as a consequence of ONNS data synchronization, autonomous TL1 notification message **`REPT-PTHCHG`** should be sent separately from a normal response to the request.

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the “Output parameters” section of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

Example input/output

The following example shows a successful completion of a **`ED-SYNCDATA-LOCAL`** command for local data synchronization operation by the NE:

```
ED-SYNCDATA-LOCAL:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
;
```

Note that data-sync is preceded by data-audit. The result of this data-audit will be stored at NN and retrievable via the command **`RTRV-AUDLOG-LOCAL`**. The result of data-audit for this case (when the modifier LOCAL is used) should be in the same format as the output for the command **`AUD-LOCAL`**.

The following example shows a successful completion of a **`ED-SYNCDATA-PATH`** command for path ID modification on local ONNS data by the NE:

```
ED-SYNCDATA-PATH:LT-ONNS-789:snn:123456:135.13.18.23-00000,135.14.196.203-32165;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: ED-SYNCDATA-PATH not possible because of local data inconsistencies

The **`ED-SYNCDATA-PATH`** command is rejected in case of local data inconsistencies with the following error response:

Error code: `E660 – LDOS – Local Data Out of Sync`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "AUD: Perform audit data" (p. 3-46)
- "RTRV-AUDLOG: Retrieve audit result log for ONNS connection data" (p. 3-2263)

ED-T3: Edit DS3 port

The command **ED-T3** is used to retrieve all parameters related to the electrical DS3 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax:

ED-T3:*tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Access Identifier of a DS3 port.
   Type: AID
   Range: APortAID of DS3 port. The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **ds3pn** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.
2. **eif** – Electrical Interface Type
   This parameter specifies the type of input to which the port is provisioned.
   Type: Alphanumeric value set.
   Range: DS3 (default), EC1
   The default value DS3 leaves the interface type unchanged, if the parameter is omitted.

3. **`fmt`** – Signal Format
   This parameter shows the format for the DS3 signal
   Type: Alphanumeric value set. A hyphen ("-") is allowed in the alphanumeric values additionally.
   Range:
   - C-BIT (default): C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format
   - UNFRAMED-CC: UNFRAMED and CLEAR-CHANNEL format
4. **`infepm`** – Incoming DS3 Far End Path PM Enable
   This parameter enables/disables the incoming DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
5. **`inlnpm`** – Incoming DS3 Line PM Enable
   This parameter enables/disables the incoming DS3 line performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
6. **`inpm`** – Incoming DS3 Near End Path PM Enable
   This parameter enables/disables the incoming DS3 near end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
7. **`lbo`** – Line Build Out
   This parameter defines the cable length applicable for the DS3 port.
   Type: Alphanumeric value set.
   Range:
   - SHORT (default): 0 … 68.5 m (0 … 225 ft)
   - LONG: 68.5 … 137 m (225 … 450 ft).
8. **`outmntc`** – Outgoing Maintenance Signal
   This parameter determines whether an AIS, a DS3AIS or IDLE signal is transmitted in response to an unconnected input.
   Type: Alphanumeric value set.
   Range: AIS (default), DS3AIS, IDLE
9. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
   This parameter enables/disables the outgoing DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).

10. **`outpm`** – Outgoing DS3 Near End Path PM Enable
This parameter enables/disables the outgoing DS3 near end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).

11. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

12. **`pmsesl`** – PM Threshold for SES – Line
PM Threshold (errors in a second) for declaring SES – Line.
Type: numeric value set.
Range: 1 … 8000
Default: 44

13. **`pmsesp`** – PM Threshold for SES – Path Incoming
PM Threshold (errors in a second) for declaring SES – Path in incoming direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44

14. **`pmsespfe`** – PM Threshold for Far-End SES – Path Incoming
PM Threshold (errors in a second) for declaring Far-End SES – Path in incoming direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44

15. **`pmsespfeout`** – PM Threshold (errors in a second) for declaring Far-End SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44

16. **`pmsespout`** – PM Threshold for SES – Path Outgoing
PM Threshold (errors in a second) for declaring SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44

17. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-3 to 10exp-9.
    Type: numeric value set.
    Range: (as exponents of 10): -4, -5 -6 (default), -7, -8, -9
18. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
19. **`tca`** – DS3 PM TCA
    The parameter tca assigns an DS3 PM TCA profile name to the designated AID.
    Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters, DEFAULT, DEFAULT0 (default).
20. **`vmr`** – Violation Monitoring Mode
    This parameter indicates the status of the DS3 P-bit parity violation monitoring and removal.
    Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.
    Range: VMR, VM, NO-VM (default)
    .

Output format

If the **`ED-T3`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **`RTRV-HDR`** command.

Detailed behavior description

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command. The modification is reported by means of an **`ED-T3`** notification for port (except for parameter eif; see below).

Switching of interface type - Ignore other parameters

If additional parameters beside *eif* are specified with the **`ED-T3`** command, they will be ignored by the NE.

Switching of interface type - Impact on port and Tributary Entities

Setting *eif* to EC1 shall

- delete the STS-1 path termination entity related to the T3 port,
- delete the T3 port entity,
- create one EC1 port entity with the defined default values for the port parameters,
- create one STS-1 tributary with the defined default values for the tributary parameters.

The deletion and creation of ports and tributaries are notified via the appropriate DLT and ENT notifications.

Switching of interface type - Impact on Fault Management

Setting *eif* to EC1 shall

- clear all active alarms for the T3 port and the related path termination
- re-evaluate the alarm status after the creation of the EC1 port and the tributary
- raise all active alarms for the EC1 port and the tributary.

The clearing and raising of alarms are notified with the appropriate notifications.

Switching of interface type - Impact on Performance Monitoring

Setting *eif* to EC1 shall

- disable all active PM monitoring points for the T3 port and the path termination
- clear all current and history bins for the T3 port and the path termination.

Example input/output

The following example shows the successful execution of a **RTRV-T3** command. Not all output parameters are listed (missing ones indicated by …).

```
RTRV-T3:LT-UNITE:1-1-#-#-21-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ”1-1-#-#-21-1:cbitmis=NORMAL,eif=DS3,ptype=T3,fmt=C-BIT,.... sigmis=NORMAL,vmr=NO
      -VM”
;
```

Error responses

See definition in “RTRV-HDR: Retrieve header” (p. 3-2417) command error responses section for general validation rules. The error responses listed there also apply to the ED-T3 command.

Note: Wrong error code when changing interface type on EP51 while XC loopback exists.

If ED-T3 command is used to change a port on EP51 pack from DS3 to EC1, while a cross-connection loopback exists at the tributary related to the port, then the command is denied with the wrong error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

The correct error code should be:

`SNVS – Status, Not in Valid State,XC loopbacks still exist`

Validation rule: Check CCB against *ptype* and *eif*

Check wether the CCB fits to *ptype* and *eif*.

Error code:`IENE – Input, Entity Not Exists`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code:`IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: *servcond* must be disabled for facility loopback

Check that the servcond is not enabled when a facility loobackp is existing for this port.

Error code:

`SNVS – Status, Not in Valid State, Facility Loopback is active for this port`

Validation rule: maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Validation rule: unframed clear-channel format not allowed with P-bit parity violation monitoring

Setting the DS3 signal format to Unframed Clear-Channel is only allowed, if the DS3 Violation Monitoring Mode parameter vmr has been set to NO-VM prior to this command or the parameter *vmr* is specified in this command with the value NO-VM.

Error code:

`SNVS – Status, Not in Valid State, Unframed Clear-Channel not allowed with P-bit parity violation monitoring`

Validation rule: P-bit parity violation monitoring not allowed with unframed clear-channel

Setting the DS3 Violation Monitoring Mode parameter vmr away from NO-VM is only allowed if the DS3 signal format has been set away from Unframed Clear-Channel prior to this command or if the parameter fmt is specified in this command with another value than UNFRAMED-CC.

Error code:

`SNVS – Status, Not in Valid State, P-bit parity violation monitoring not allowed with Unframed Clear-Channel`

Validation rule: DS3 path PM not allowed with unframed clear-channel

Setting the parameters *inpm* and *outpm* to ENABLE is only alowed if the DS3 signal format has been set away from Unframed Clear-Channel prior to this command or if the parameter *fmt* is specified in this command with another value than UNFRAMED-CC.

Error code:

`SNVS – Status, Not in Valid State, DS3 path PM not allowed with Unframed Clear-Channel`

Validation rule: unframed clear-channel not allowed with DS3 path PM

Setting the DS3 signal format to Unframed Clear-Channel is only allowed, if the DS3 path PM has been disabled prior to this command of if both parameters *inpm* and *outpm* are specified in this command with the value DISABLE.

Error code:

`SNVS – Status, Not in Valid State, Unframed Clear-Channel not allowed with DS3 path PM`

Validation rule: DS3AIS or IDLE not allowed with unframed clear-channel

Setting the parameter *outmntc* to DS3AIS or IDLE is not allowed if the DS3 signal format has been set to Unframed Clear-Channel prior to this command or if the parameter *fmt* is specified in this command with UNFRAMED-CC.

Error code:

`SNVS – Status, Not in Valid State, DS3AIS or IDLE not allowed with Unframed Clear-Channel`

Validation rule: no cross-connections for interface type change

Check that eif is not changed when a facility loopback is still on this port.

Error code:

`SNVS – Status, Not in Valid State, cross-connections still existing`

Validation rule: no conversion cross-connections for interface type change

Check that *eif* is not changed when conversion cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: No facility loopbacks for interface type change

Check that *eif* is not changed when cross-connections are still on this port.

Error code:

`SNVS – Status, Not in Valid State, facility loopbacks still exist`

Validation rule: no XC loopbacks for interface type change

Check that *eif* is not changed when cross-connections loopback is still on this port.

Error code:

`SNVS – Status, Not in Valid State, a cross-connect loopback exists`

Validation rule: no test access session for interface type change

Check that *eif* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Check whether system is in maintenance condition

Check that eif is not changed when the system in is maintenance condition.

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Related TL1 messages

"RTRV-T3: Retrieve DS3 port" (p. 3-2968)

References

GR-199-CORE (RTRV-*rr*)

ED-T3 notification

Output format

Syntax

`ED-T3`:*aid::spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range: APortAID of DS3 port

Spec block parameters

1. **`ds3pn`** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.
2. **`eif`** – Electrical Interface Type
   The parameter eif indicates the type of input the port has been provisioned to receive.
   Type: Alphanumeric value set.
   Range: DS3
3. **`fmt`** – Signal Format
   This parameter shows the format for the DS3 signal
   Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.

   Range:
   - C-BIT (default): C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format
   - UNFRAMED-CC: UNFRAMED and CLEAR-CHANNEL format
4. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

Range:

- YES
- NO (default).

5. **`infepm`** – Incoming DS3 Far End Path PM Enable
   This parameter enables/disables the incoming DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
6. **`inlnpm`** – Incoming DS3 Line PM Enable
   This parameter enables/disables the incoming DS3 line performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
7. **`inpm`** – Incoming DS3 Near End Path PM Enable
   This parameter enables/disables the incoming DS3 near end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
8. **`lbo`** – Line Build Out
   This parameter defines the cable length applicable for the DS3 port.
   Type: Alphanumeric value set.
   Range:
   - SHORT (default): 0 … 68.5 m (0 … 225 ft)
   - LONG: 68.5 … 137 m (225 … 450 ft).
9. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
10. **`outmntc`** – Outgoing Maintenance Signal
    This parameter determines whether an AIS, a DS3AIS or IDLE signal is transmitted in response to an unconnected input.
    Type: Alphanumeric value set.
    Range: AIS (default), DS3AIS, IDLE

11. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
    This parameter enables/disables the outgoing DS3 far end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
12. **`outpm`** – Outgoing DS3 Near End Path PM Enable
    This parameter enables/disables the outgoing DS3 near end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
13. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.
    Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
14. **`pmsesl`** – PM Threshold for SES – Line
    PM Threshold (errors in a second) for declaring SES – Line
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
15. **`pmsesp`** – PM Threshold for SES – Path Incoming
    PM Threshold (errors in a second) for declaring SES – Path in incoming direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
16. **`pmsespfe`** – PM Threshold for Far-End SES – Path Incoming
    PM Threshold (errors in a second) for declaring Far-End SES – Path in incoming direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
17. **`pmsespfeout`** – PM Threshold (errors in a second) for declaring Far-End SES – Path in outgoing direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44

18. **`pmsespout`** – PM Threshold for SES – Path Outgoing
PM Threshold (errors in a second) for declaring SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44
19. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-3 to 10exp-9.
Type: numeric value set.
Range: (as exponents of 10): -4, -5 -6 (default), -7, -8, -9
20. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
21. **`tca`** – DS3 PM TCA
The parameter tca assigns an DS3 PM TCA profile name to the designated AID.
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters, DEFAULT, DEFAULT0 (default).
22. **`vmr`** – Violation Monitoring Mode
This parameter indicates the status of the DS3 P-bit parity violation monitoring and removal.
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.
Range: VMR, VM, NO-VM (default)
.

Example input/output

```
ED-T3:UNITE-NE:1-1-#-#-21-1:123456:::FMT=UNFRAMED-CC,INPM=DISABLE,OUTPM=DISABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ED-T3:1-1-#-#-21-1::fmt=unframed-
      cc,inpm=disable,outpm=disable:″
;
```

ED-TCA-PROF: Edit TCA profile

General

The **ED-TCA-PROF** command allows the user to modify the existing performance monitoring threshold crossing alert (TCA) profile contents or the profile name. The system default profile names (Default and Default0) cannot be changed. For a general description of TCA profiles, refer to the **RTRV-TCA-PROF** command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM3

Priority: 1

Abortable: No

Input format

Command syntax: **ED-TCA-PROF**:*tid::ctag::pftype,pfname:spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
4. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value.
   NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Spec block parameter

1. newname
   This parameter allows a new name to be assigned to the already established profile.
   Type: Alphanumeric string.
   Range: This is a alphanumeric character string with a maximum of 24 characters. Empty string is not allowed.

Spec block parameters - profile parameters of type SECTION-LINE

1. **qhcvsoc3**
   15-minute threshold for CV-S bin for OC-3.
   Type: Integer.
   Range: 0-57599100
   Default: 140.
2. **daycvsoc3**
   Day threshold for CV-S bin of OC-3.
   Type: Integer.
   Range: 0-4294967295
   Default: 1344.
3. **qhcvsoc12**
   15-minute threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-57599100
   Default: 560.
4. **daycvsoc12**
   Day threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-4294967295
   Default: 5376.
5. **qhcvsoc48**
   15-minute threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-57599100
   Default: 2240.
6. **daycvsoc48**
   Day threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-4294967295
   Default: 21504.
7. **qhcvsoc192**
   15-minute threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-57599100
   Default: 8960.

8. **daycvsoc192**
   Day threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-4294967295
   Default: 86016.
9. **qhcvsoc768**
   15-minute threshold for CV-S bin for OC-768.
   Type: Integer.
   Range: 0-57599100
   Default: 35840.
10. **daycvsoc768**
   Day threshold for CV-S bin for OC-768.
   Type: Integer.
   Range: 0-4294967295
   Default: 344064.
11. **qhcvsec1**
   15-minute threshold for CV-S bin for EC-1.
   Type: Integer.
   Range: 0-57599100
   Default: 47
12. **daycvsec1**
   Day threshold for CV-S bin of EC-1.
   Type: Integer.
   Range: 0-4294967295
   Default: 448
13. **qhess**
   15-minute threshold for ES-S bin.
   Type: Integer.
   Range: 0-900
   Default: 25.
14. **dayess**
   Day threshold for ES-S bin.
   Type: Integer.
   Range: 0-86400
   Default: 250.
15. **qhsess**
   15-minute threshold for SES-S bin.
   Type: Integer.
   Range: 0-810
   Default: 10.

16. **`daysess`**
Day threshold for ES-S bin.
Type: Integer.
Range: 0-77760
Default: 40.

17. **`qhsefss`**
15-minute threshold for SEFS-S bin.
Type: Integer.
Range: 0-900
Default: 5.

18. **`daysefss`**
Day threshold for SEFS-S bin.
Type: Integer.
Range: 0-86400
Default: 10.

19. **`qhcvloc3`**
15-minute threshold for CV-L and CV-LFE bin for OC-3.
Type: Integer.
Range: 0-172799100
Default: 140.

20. **`daycvloc3`**
Day threshold for CV-L bin and CV-LFE for OC-3.
Type: Integer.
Range: 0-4294967295
Default: 1344.

21. **`qhcvloc12`**
15-minute threshold for CV-L and CV-LFE bin for OC-12.
Type: Integer.
Range: 0-691199100
Default: 560.

22. **`daycvloc12`**
Day threshold for CV-L bin and CV-LFE for OC-12.
Type: Integer.
Range: 0-4294967295
Default: 5376.

23. **`qhcvloc48`**
15-minute threshold for CV-L and CV-LFE bin for OC-48.
Type: Integer.
Range: 0-4294967295
Default: 2240.

24. **`daycvloc48`**
    Day threshold for CV-L and CV-LFE bin for OC-48.
    Type: Integer.
    Range: 0-4294967295
    Default: 21504.
25. **`qhcvloc192`**
    15-minute threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 8960.
26. **`daycvloc192`**
    Day threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 86016.
27. **`qhcvloc768`**
    15-minute threshold for CV-L and CV-LFE bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 35840.
28. **`daycvloc768`**
    Day threshold for CV-L and CV-LFE bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 344064.
29. **`qhcvlec1`**
    15-minute threshold for CV-L and CV-LFE bin for EC-1.
    Type: Integer.
    Range: 0-57599100
    Default: 47
30. **`daycvlec1`**
    Day threshold for CV-L bin and CV-LFE for EC-1.
    Type: Integer.
    Range: 0-4294967295
    Default: 448
31. **`qhesl`**
    15-minute threshold for ES-L and ES-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 25.

32. **`dayesl`**
    Day threshold for ES-L and ES-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 250.
33. **`qhfeccloc192`**
    15-minute threshold for FECC-L bin for OC-192.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 5184.
34. **`dayfeccloc192`**
    Day threshold for FECC-L bin for OC-192.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 497644.
35. **`qhfeccloc768`**
    15-minute threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
36. **`dayfeccloc768`**
    Day threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
37. **`qhsesl`**
    15-minute threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
38. **`daysesl`**
    Day threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.

39. **qhuasl**
    15-minute threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.

40. **dayuasl**
    Day threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

Spec block parameters - profile parameters of type RS-MS

1. **qhrsnbbe**
   15-minute threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 9000.

2. **dayrsnbbe**
   Day threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

3. **qhrsnes**
   15-minute threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 180.

4. **dayrsnes**
   Day threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.

5. **qhrsnses**
   15-minute threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 15.

6. **dayrsnses**
   Day threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 20.

7. **qhrsnuas**
   15-minute threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayrsnuas**
   Day threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 20.
9. **qhmsbbestm1**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 228000.
10. **daymsbbestm1**
   Day threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 432000.
11. **qhmsbbestm4**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 0-691199100
   Default: 912000.
12. **daymsbbestm4**
   Day threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 00-4294967295
   Default: 1728000.
13. **qhmsbbestm16**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
   Type: Integer.
   Range: 0-2764799100
   Default: 3648000.
14. **daymsbbestm16**
   Day threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 6912000.

15. **qhmsbbestm64**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 14592000.
16. **daymsbbestm64**
    Day threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 27648000.
17. **qhmsbbestm256**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 58368000.
18. **daymsbbestm256**
    Day threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 110592000.
19. **qhmses**
    15-minute threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 50.
20. **daymses**
    Day threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.
21. **qhmsnfeccstm64**
    15-minute threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 5184.
22. **daymsnfeccstm64**
    Day threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 497644.

23. **`qhmsnfeccstm256`**
    15-minute threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
24. **`daymsnfeccstm256`**
    Day threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
25. **`qhmsses`**
    15-minute threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
26. **`daymsses`**
    Day threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-77760
    Default: 15.
27. **`qhmsuas`**
    15-min threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
28. **`daymsuas`**
    Day threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters - reset profile parameters of type RS-MS

1. **`qhrtrrsnbbe`**
   15-minute reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **`dayrtrrsnbbe`**
   Day reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.

3. **qhrtrrsnes**
   15-minute reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 20.
4. **dayrtrrsnes**
   Day reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **qhrtrrsnses**
   15-minute reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrrsnses**
   Day reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 0.
7. **qhrtrrsnuas**
   15-minute reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **dayrtrrsnuas**
   Day reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 0.
9. **qhrtrmsbbestm1**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 2400.
10. **dayrtrmsbbestm1**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 43200.

11. **qhrtrmsbbestm4**
15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
Type: Integer.
Range: 0-691199100
Default: 9600.
12. **dayrtrmsbbestm4**
Day reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
Type: Integer.
Range: 0-4294967295
Default: 172800.
13. **qhrtrmsbbestm16**
15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
Type: Integer.
Range: 0-2764799100
Default: 38400.
14. **dayrtrmsbbestm16**
Day reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
Type: Integer.
Range: 0-4294967295
Default: 691200.
15. **qhrtrmsbbestm64**
15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
Type: Integer.
Range: 0-4294967295
Default: 153600.
16. **dayrtrmsbbestm64**
Day reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
Type: Integer.
Range: 0-4294967295
Default: 2764800.
17. **qhrtrmsbbestm256**
15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
Type: Integer.
Range: 0-4294967295
Default: 614400.
18. **dayrtrmsbbestm256**
Day reset threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
Type: Integer.
Range: 0-4294967295
Default: 11059200.

19. **qhrtrmses**
    15-minute reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
20. **dayrtrmses**
    Day reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 15.
21. **qhrtrmsnfeccstm64**
    15-minute reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
22. **dayrtrmsnfeccstm64**
    Day reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
23. **qhrtrmsnfeccstm256**
    15-minute reset threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
24. **dayrtrmsnfeccstm256**
    Day reset threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
25. **qhrtrmsses**
    15-minute reset threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
26. **dayrtrmsses**
    Day reset threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.

27. **qhrtrmsuas**
15-min reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-900
Default: 0.
28. **dayrtrmsuas**
Day reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-86400
Default: 0.

Spec block parameters - profile parameters of type ODU

1. **qhodupbbeodu1**
15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-18377100
Default: 32000.
2. **dayodupbbeodu1**
Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-1764201600
Default: 61000.
3. **qhodupes**
15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
Type: Integer.
Range: 0-900
Default: 50.
4. **dayodupes**
Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P..
Type: Integer.
Range: 0-86400
Default: 150.
5. **qhodupses**
15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
Type: Integer.
Range: 0-810
Default: 10.
6. **dayodupses**
Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
Type: Integer.
Range: 0-77760
Default: 15.

7. **`qhodupuas`**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **`dayodupuas`**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 20.

Spec block parameters - reset profile parameters of type ODU

1. **`qhrtrodupbbeodu1`**
   15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-18377100
   Default: 200.
2. **`dayrtrodupbbeodu1`**
   Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-1764201600
   Default: 4800.
3. **`qhrtrodupes`**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 5.
4. **`dayrtrodupes`**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 15.
5. **`qhrtrodupses`**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **`dayrtrodupses`**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 0.

7. **qhrtrodupuas**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **dayrtrodupuas**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 0.

Spec block parameters - profile parameters of type HO-VC

1. **qhvcbbevc3**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
2. **dayvcbbevc3**
   Day threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
3. **qhvcbbevc4**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbevc4**
   Day threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcbbevc44c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
6. **dayvcbbevc44c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

7. **qhvcbbevc416c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
8. **dayvcbbevc416c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
9. **qhvcbbevc464c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
10. **dayvcbbevc464c**
    Day threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
    Type: Integer.
    Range: 0-691113600
    Default: 48000.
11. **qhvcesvc3**
    15-minute threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 150.
12. **dayvcesvc3**
    Day threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 600.
13. **qhvcesvc4**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
14. **dayvcesvc4**
    Day threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.

15. **qhvcesvc44c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-900
Default: 180.
16. **dayvcesvc44c**
Day threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-86400
Default: 1500.
17. **qhvcesvc416c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-900
Default: 180.
18. **dayvcesvc416c**
Day threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-86400
Default: 1500.
19. **qhvcesvc464c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 180.
20. **dayvcesvc464c**
Day threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 1500.
21. **qhvcsesvc3**
15-minute threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-810
Default: 15.
22. **dayvcsesvc3**
Day threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-77760
Default: 20.

23. **qhvcsesvc4**
    15-minute threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
24. **dayvcsesvc4**
    Day threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
25. **qhvcsesvc44c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
26. **dayvcsesvc44c**
    Day threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Range: 0-77760
    Default: 20.
27. **qhvcsesvc416c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
28. **dayvcsesvc416c**
    Day threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
29. **qhvcsesvc464c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
30. **dayvcsesvc464c**
    Day threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.

31. **qhvcuasvc3**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
32. **dayvcuasvc3**
    Day threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
33. **qhvcuasvc4**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
34. **dayvcuasvc4**
    Day threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
35. **qhvcuasvc44c**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
36. **dayvcuasvc44c**
    Day threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
37. **qhvcuasvc416c**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
38. **dayvcuasvc416c**
    Day threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

39. **`qhvcuasvc464c`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
40. **`dayvcuasvc464c`**
    Day threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
41. **`qhpjcpos`**
    15-minute threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
42. **`daypjcpos`**
    Day threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
43. **`qhpjcneg`**
    15-minute threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
44. **`daypjcneg`**
    Day threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Spec block parameters - reset profile parameters of type HO-VC

1. **`qhrtrvcbbevc3`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **`dayrtrvcbbevc3`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.

3. **qhrtrvcbbevc4**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **dayrtrvcbbevc4**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
5. **qhrtrvcbbevc44c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
6. **dayrtrvcbbevc44c**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
7. **qhrtrvcbbevc416c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
8. **dayrtrvcbbevc416c**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
9. **qhrtrvcbbevc464c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
10. **dayrtrvcbbevc464c**
    Day reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
    Type: Integer.
    Range: 0-691113600
    Default: 4800.

11. **qhrtrvcesvc3**
    15-minute reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
12. **dayrtrvcesvc3**
    Day reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 60.
13. **qhrtrvcesvc4**
    15-minute reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 20.
14. **dayrtrvcesvc4**
    Day reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.
15. **qhrtrvcesvc44c**
    15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
    Type: Integer.
    Range: 0-900
    Default: 20.
16. **dayrtrvcesvc44c**
    Day reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.
17. **qhrtrvcesvc416c**
    15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-900
    Default: 20.
18. **dayrtrvcesvc416c**
    Day reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.

19. **qhrtrvcesvc464c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 20.
20. **dayrtrvcesvc464c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 150.
21. **qhrtrvcsesvc3**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-810
Default: 0.
22. **dayrtrvcsesvc3**
Day reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-77760
Default: 0.
23. **qhrtrvcsesvc4**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-810
Default: 0.
24. **dayrtrvcsesvc4**
Day reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-77760
Default: 0.
25. **qhrtrvcsesvc44c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
Type: Integer.
Range: 0-810
Default: 0.
26. **dayrtrvcsesvc44c**
Day reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
Type: Integer.
Range: 0-77760
Default: 0.

27. **qhrtrvcsesvc416c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-810
Default: 0.
28. **dayrtrvcsesvc416c**
Day reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-77760
Default: 0.
29. **qhrtrvcsesvc464c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-810
Default: 0.
30. **dayrtrvcsesvc464c**
Day reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-77760
Default: 0.
31. **qhrtrvcuasvc3**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-900
Default: 0.
32. **dayrtrvcuasvc3**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-86400
Default: 0.
33. **qhrtrvcuasvc4**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-900
Default: 0.
34. **dayrtrvcuasvc4**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 0.

35. **qhrtrvcuasvc44c**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
Type: Integer.
Range: 0-900
Default: 0.
36. **dayrtrvcuasvc44c**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
Type: Integer.
Range: 0-86400
Default: 0.
37. **qhrtrvcuasvc416c**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
Type: Integer.
Range: 0-900
Default: 0.
38. **dayrtrvcuasvc416c**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
Type: Integer.
Range: 0-86400
Default: 0.
39. **qhrtrvcuasvc464c**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
Type: Integer.
Range: 0-900
Default: 0.
40. **dayrtrvcuasvc464c**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
Type: Integer.
Range: 0-86400
Default: 0.
41. **qhrtrpjcpos**
15-minute reset threshold for positive PJC bin.
Type: Integer.
Range: 0-1800000
Default: 0.
42. **dayrtrpjcpos**
Day reset threshold for positive PJC bin.
Type: Integer.
Range: 0-172800000
Default: 0.

43. **`qhrtrpjcneg`**
    15-minute reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
44. **`dayrtrpjcneg`**
    Day reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.

Spec block parameters - profile parameters of type PATH

1. **`qhcvpsts1`**
   15-minute threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-57599100
   Default: 15.
2. **`daycvpsts1`**
   Day threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-4294967295
   Default: 125.
3. **`qhcvpsts3c`**
   15-minute threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 0-57599100
   Default: 25.
4. **`daycvpsts3c`**
   Day threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 4294967295
   Default: 250.
5. **`qhcvpsts12c`**
   15-minute threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 0-57599100
   Default: 75.
6. **`daycvpsts12c`**
   Day threshold for threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 4294967295
   Default: 750.

7. **qhcvpsts48c**
   15-minute threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 0-57599100
   Default: 300.
8. **daycvpsts48c**
   Day threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 4294967295
   Default: 3000.
9. **qhcvpsts192c**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-57599100
   Default: 1200.
10. **daycvpsts192c**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-4294967295
   Default: 12000.
11. **qhespsts1**
   15-minute threshold for ES-P and ES-PFE bin for STS1.
   Type: Integer.
   Range: 0-900
   Default: 12.
12. **dayespsts1**
   Day threshold for ES-P and ES-PFE bin for STS1.
   Type: Integer.
   Range: 0-86400
   Default: 100.
13. **qhespsts3c**
   15-minute threshold for ES-P and ES-PFE bin for STS3C.
   Type: Integer.
   Range: 0-900
   Default: 20.
14. **dayespsts3c**
   Day threshold for ES-P and ES-PFE bin for STS3C.
   Type: Integer.
   Range: 0-86400
   Default: 200.

15. **qhespsts12c**
    15-minute threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-900
    Default: 60.
16. **dayespsts12c**
    Day threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-86400
    Default: 600.
17. **qhespsts48c**
    15-minute threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-900
    Default: 240.
18. **dayespsts48c**
    Day threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-86400
    Default: 2400.
19. **qhespsts192c**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-900
    Default: 240.
20. **dayespsts192c**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-86400
    Default: 9600.
21. **qhsesp**
    15-minute threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-810
    Default: 3.
22. **daysesp**
    Day threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 7.

23. **qhuasp**
    15-minute threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
24. **dayuasp**
    Day threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.
25. **qhppjcgen**
    15-minute threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
26. **dayppjcgen**
    Day threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
27. **qhnpjcgen**
    15-minute threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
28. **daynpjcgen**
    Day threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
29. **qhppjcgdet**
    15-minute threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
30. **dayppjcdet**
    Day threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

31. **`qhnpjcdet`**
    15-minute threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
32. **`daynpjcdet`**
    Day threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Spec block parameters - profile parameters of type ENET

1. **`qhpdege`**
   15-minute threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 187500.
2. **`daypdege`**
   Day threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.
3. **`qhpdevcg`**
   15-minute threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 10.
4. **`daypdevcg`**
   Day threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
5. **`qhedfe1ge`**
   15min EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 1875000.
6. **`dayedfe1ge`**
   24 h EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.

7. **`qhedfc1ge`**
   15min EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 20.
8. **`dayedfc1ge`**
   24 h EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
9. **`qhedfe10ge`**
   15min EDFE for 10GbE Ethernet port.
   Type: Integer.
   Range: 0-2147483647
   Default: 18750000.
10. **`dayedfe10ge`**
    24 h EDFE for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 90000000.
11. **`qhedfc10ge`**
    15min EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
12. **`dayedfc10ge`**
    24 h EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.
13. **`qhedfe1gevcg`**
    15min EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
14. **`dayedfe1gevcg`**
    24 h EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

15. **`qhedfc1gevcg`**
    15min EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
16. **`dayedfc1gevcg`**
    24 h EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
17. **`qhedfe10gevcg`**
    15min EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
18. **`dayedfe10gevcg`**
    24 h EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.
19. **`qhedfc10gevcg`**
    15min EDFC for 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
20. **`dayedfc10gevcg`**
    24 h EDFC for 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.

Spec block parameters - reset profile parameters of type ENET

1. **`qhrtrpdege`**
   15-minute reset threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 937500.
2. **`dayrtrpdege`**
   Day reset threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 4500000.

3. **`qhrtrpdevcg`**
   15-minute reset threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 2.
4. **`dayrtrpdevcg`**
   Day reset threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
5. **`qhrtredfe1ge`**
   15min reset threshold for EDFE of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 937500.
6. **`dayrtredfe1ge`**
   24h reset threshold for EDFE of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 4500000.
7. **`qhrtredfc1ge`**
   15min reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 2.
8. **`dayrtredfc1ge`**
   24h reset threshold for EDFC of 1GbE Ethernet por.
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
9. **`qhrtredfe10ge`**
   15min reset threshold for EDFE of 10GbE Ethernet port.
   Type: Integer.
   Range: 0-2147483647
   Default: 9375000
10. **`dayrtredfe10ge`**
    24h reset threshold for EDFE of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 45000000.

11. **`qhrtredfc10ge`**
    15min reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
12. **`dayrtredfc10ge`**
    24h reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
13. **`qhrtredfe1gevcg`**
    15min reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
14. **`dayrtredfe1gevcg`**
    24h reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
15. **`qhrtredfc1gevcg`**
    15min reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
16. **`dayrtredfc1gevcg`**
    24h reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
17. **`qhrtredfe10gevcg`**
    15min reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
18. **`dayrtredfe10gevcg`**
    24h reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

19. **`qhrtredfc10gevcg`**
    15min reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
20. **`dayrtredfc10gevcg`**
    24h reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

Spec block parameters- profile parameters of type LOVC

1. **`qhvcbbevc12`**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 9000.
2. **`dayvcbbevc12`**
   Day threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 12000.
3. **`qhvcbbelovc3`**
   15-minute threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **`dayvcbbelovc3`**
   Day threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **`qhvcesvc12`**
   15-minute threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 120.
6. **`dayvcesvc12`**
   Day threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 350.

7. **qhvceslovc3**
   15-minute threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 150.
8. **dayvceslovc3**
   Day threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 600.
9. **qhvcsesvc12**
   15-minute threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
10. **dayvcsesvc12**
    Day threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
11. **qhvcseslovc3**
    15-minute threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
12. **dayvcseslovc3**
    Day threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
13. **qhvcuasvc12**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
14. **dayvcuasvc12**
    Day threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

15. **`qhvcuaslovc3`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
16. **`dayvcuaslovc3`**
    Day threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters- reset profile parameters of type LOVC

1. **`qhrtrvcbbevc12`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 50.
2. **`dayrtrvcbbevc12`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 1200.
3. **`qhrtrvcbbelovc3`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **`dayrtrvcbbelovc3`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
5. **`qhrtrvcesvc12`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 5.
6. **`dayrtrvcesvc12`**
   Day reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 35.

7. **`qhrtrvceslovc3`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 10.
8. **`dayrtrvceslovc3`**
   Day reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 60.
9. **`qhrtrvcsesvc12`**
   15-minute reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
10. **`dayrtrvcsesvc12`**
    Day reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
11. **`qhrtrvcseslovc3`**
    15-minute reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
12. **`dayrtrvcseslovc3`**
    Day reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
13. **`qhrtrvcuasvc12`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
14. **`dayrtrvcuasvc12`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

15. **qhrtrvcuaslovc3**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.

16. **dayrtrvcuaslovc3**
    Day reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

Spec block parameters.- Profile Parameters of Type LOPATH

1. **qhcvvt1**
   15-minute threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-14399100
   Default: 75.

2. **daycvvt1**
   Day threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-1382313600
   Default: 750.

3. **qhesvt1**
   15-minute threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.

4. **dayesvt1**
   Day threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.

5. **qhsesvt1**
   15-minute threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-810
   Default: 40.

6. **daysesvt1**
   Day threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-77760
   Default: 900.

7. **qhuasvt1**
   15-minute threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
8. **dayuasvt1**
   Day threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.

Command parameters - profile parameters of type DS3

1. **qhcvl**
   15-minute threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0-7200000
   Default: 387.
2. **daycvl**
   Day threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0- 691200000
   Default: 3865.
3. **qhesl**
   15-minute threshold for Es-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
4. **dayesl**
   Day threshold for ES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
5. **qhesal**
   15-minute threshold for ESA-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
6. **dayesal**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.

7. **qhesbl**
   15-minute threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
8. **dayesbl**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
9. **qhsesl**
   15-minute threshold for SES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-810
   Default: 4.
10. **daysesl**
   Day threshold for SES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-77760
   Default: 40.
11. **qhcvp**
   15-minute threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
   Type: Integer.
   Range: 0-7200000
   Default: 382.
12. **daycvp**
   Day threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
   Type: Integer.
   Range: 0-691200000
   Default: 3820.
13. **qhesp**
   15-minute threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
   Type: Integer.
   Range: 0-900
   Default: 25.
14. **dayesp**
   Day threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 250.

15. **qhesap**
    15-minute threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
16. **dayesap**
    Day threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
17. **qhesbp**
    15-minute threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
18. **dayesbp**
    Day threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
19. **qhsesp**
    15-minute threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-810
    Default: 4.
20. **daysesp**
    Day threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-77760
    Default: 40.
21. **qhuasp**
    15-minute threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 10.

22. **`dayuasp`**
    Day threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 10.
23. **`qhsasp`**
    15-minute threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
24. **`daysasp`**
    Day threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 8.

Command parameters - profile parameters of type DS1

1. **`qhcvsfp`**
   15-minute threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 72.
2. **`daycvsfp`**
   Day threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0- 1048575
   Default: 691.
3. **`qhcvesfp`**
   15-minute threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 13296.
4. **`daycvesfp`**
   Day threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-1048575
   Default: 132960.

5. **qhesp**
   15-minute threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 65.
6. **dayesp**
   Day threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 648.
7. **qhsesp**
   15-minute threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-810
   Default: 10.
8. **daysesp**
   Day threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 250.
9. **qhuasp**
   15-minute threshold for UAS-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 10.
10. **dayuasp**
    Day threshold for UAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 10.
11. **qhsasp**
    15-minute threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
12. **daysasp**
    Day threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 17.

Output format

If the NE fully complies with the ED-TCA-PROF request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

TCA Profile modification

The command modifies the existing performance monitoring Threshold Crossing Alert (TCA) profile content or the profile name.

The modification of the TCA profile is reported by means of an ED-TCA-PROF notification.

Example input/output

The following example shows the successful completion of a ED-TCA-PROF command by the NE:

```
ED-TCA-PROF:UNITE-NE::123456::rs-ms,ATT:qhrsnes=10,qhrsnbbe=6;
   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`ED-TCA-PROF`** command.

Validation rule: profile name does not exist

If the profile name specified does not exist.

Error code: `IDNV – Input, Data Not Valid, profile name does not exist`

Validation rule: profile name already exists

If the system receives a new profile name that already exists for that profile type, the following error response is returned.

Error code: `IEAE – Input, Entity Already Exists, duplicate profile name`

Validation rule: rename default profile name

If a user tries to change the system default profile name (Default / Default0), the following error response is returned.

Error code:

`IDNC – Input, Data Not Consistent, new name for default profile`

Validation rule: value change of default profile Default0

If a user tries to change values of the system default profile ″Default0″, the following error response is returned.

Error code:

`IDNC – Input, Data Not Consistent, value change not allowed for profile Default0`

Related TL1 messages

- “DLT-TCA-PROF: Delete TCA profile” (p. 3-259)
- “RTRV-TCA-PROF: Retrieve TCA profile” (p. 3-2982)
- “ENT-TCA-PROF: Enter TCA Profile” (p. 3-1894)

References

GR-833-CORE

ED-TCA-PROF notification

Output format

Command syntax: **`ED-TCA-PROF`**::*pftype*,*pfname*:*spec_block*

Output parameter

Command parameters

1. **`pftype`** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
2. **`pfname`** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **`ENT-TCA-PROF`**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value.
   NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Spec block parameter

1. **`newname`**
   This parameter allows a new name to be assigned to the already established profile.
   Type: Alphanumeric string.
   Range: This is a alphanumeric character string with a maximum of 24 characters.
   Empty string is not allowed.

Spec block parameters - profile parameters of type SECTION-LINE

1. **`qhcvsoc3`**
   15-minute threshold for CV-S bin for OC-3.
   Type: Integer.
   Range: 0-57599100
   Default: 140.
2. **`daycvsoc3`**
   Day threshold for CV-S bin of OC-3.
   Type: Integer.
   Range: 0-4294967295
   Default: 1344.

3. **qhcvsoc12**
   15-minute threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-57599100
   Default: 560.
4. **daycvsoc12**
   Day threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-4294967295
   Default: 5376.
5. **qhcvsoc48**
   15-minute threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-57599100
   Default: 2240.
6. **daycvsoc48**
   Day threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-4294967295
   Default: 21504.
7. **qhcvsoc192**
   15-minute threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-57599100
   Default: 8960.
8. **daycvsoc192**
   Day threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-4294967295
   Default: 86016.
9. **qhcvsoc768**
   15-minute threshold for CV-S bin for OC-768.
   Type: Integer.
   Range: 0-57599100
   Default: 35840.
10. **daycvsoc768**
    Day threshold for CV-S bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 344064.

11. **`qhcvsec1`**
    15-minute threshold for CV-S bin for EC-1.
    Type: Integer.
    Range: 0-57599100
    Default: 47
12. **`daycvsec1`**
    Day threshold for CV-S bin of EC-1.
    Type: Integer.
    Range: 0-4294967295
    Default. 448
13. **`qhess`**
    15-minute threshold for ES-S bin.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **`dayess`**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **`qhsess`**
    15-minute threshold for SES-S bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
16. **`daysess`**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.
17. **`qhsefss`**
    15-minute threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
18. **`daysefss`**
    Day threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

19. **qhcvloc3**
 15-minute threshold for CV-L and CV-LFE bin for OC-3.
 Type: Integer.
 Range: 0-172799100
 Default: 140.
20. **daycvloc3**
 Day threshold for CV-L bin and CV-LFE for OC-3.
 Type: Integer.
 Range: 0-4294967295
 Default: 1344.
21. **qhcvloc12**
 15-minute threshold for CV-L and CV-LFE bin for OC-12.
 Type: Integer.
 Range: 0-691199100
 Default: 560.
22. **daycvloc12**
 Day threshold for CV-L bin and CV-LFE for OC-12.
 Type: Integer.
 Range: 0-4294967295
 Default: 5376.
23. **qhcvloc48**
 15-minute threshold for CV-L and CV-LFE bin for OC-48.
 Type: Integer.
 Range: 0-2764799100
 Default: 2240.
24. **daycvloc48**
 Day threshold for CV-L and CV-LFE bin for OC-48.
 Type: Integer.
 Range: 0-4294967295
 Default: 21504.
25. **qhcvloc192**
 15-minute threshold for CV-L and CV-LFE bin for OC-192.
 Type: Integer.
 Range: 0-4294967295
 Default: 8960.
26. **daycvloc192**
 Day threshold for CV-L and CV-LFE bin for OC-192.
 Type: Integer.
 Range: 0-4294967295
 Default: 86016.

27. **`qhcvloc768`**
15-minute threshold for CV-L and CV-LFE bin for OC-768.
Type: Integer.
Range: 0-4294967295
Default: 35840.

28. **`daycvloc768`**
Day threshold for CV-L and CV-LFE bin for OC-768.
Type: Integer.
Range: 0-4294967295
Default: 344064.

29. **`qhcvlec1`**
15-minute threshold for CV-L and CV-LFE bin for EC-1.
Type: Integer.
Range: 0-57599100
Default: 47

30. **`daycvlec1`**
Day threshold for CV-L bin and CV-LFE for EC-1.
Type: Integer.
Range: 0-4294967295
Default: 448

31. **`qhesl`**
15-minute threshold for ES-L and ES-LFE bin.
Type: Integer.
Range: 0-900
Default: 25.

32. **`dayesl`**
Day threshold for ES-L and ES-LFE bin.
Type: Integer.
Range: 0-86400
Default: 250.

33. **`qhfeccloc192`**
15-minute threshold for FECC-L bin for OC-192.
Type: Integer.
Range: 0 – 4294967295
Default: 5184.

34. **`dayfeccloc192`**
Day threshold for FECC-L bin for OC-192.
Type: Integer.
Range: 0 – 4294967295
Default: 497644.

35. **`qhfeccloc768`**
    15-minute threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
36. **`dayfeccloc768`**
    Day threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
37. **`qhsesl`**
    15-minute threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
38. **`daysesl`**
    Day threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.
39. **`qhuasl`**
    15-minute threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
40. **`dayuasl`**
    Day threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

Spec block parameters - profile parameters of type RS-MS

1. **`qhrsnbbe`**
   15-minute threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 9000.
2. **`dayrsnbbe`**
   Day threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

3. **qhrsnes**
   15-minute threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 180.
4. **dayrsnes**
   Day threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.
5. **qhrsnses**
   15-minute threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
6. **dayrsnses**
   Day threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 20.
7. **qhrsnuas**
   15-minute threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayrsnuas**
   Day threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 20.
9. **qhmsbbestm1**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 228000.
10. **daymsbbestm1**
   Day threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 432000.

11. **qhmsbbestm4**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 0-691199100
    Default: 912000.
12. **daymsbbestm4**
    Day threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 00-4294967295
    Default: 1728000.
13. **qhmsbbestm16**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-2764799100
    Default: 3648000.
14. **daymsbbestm16**
    Day threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 6912000.
15. **qhmsbbestm64**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 14592000.
16. **daymsbbestm64**
    Day threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 27648000.
17. **qhmsbbestm256**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 58368000.
18. **daymsbbestm256**
    Day threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 110592000.

19. **qhmses**
 15-minute threshold for MS_N_ES and MS_F_ES bin.
 Type: Integer.
 Range: 0-900
 Default: 50.
20. **daymses**
 Day threshold for MS_N_ES and MS_F_ES bin.
 Type: Integer.
 Range: 0-86400
 Default: 150.
21. **qhmsnfeccstm64**
 15-minute threshold for MS-N-FECC bin forSTM-64.
 Type: Integer.
 Range: 0 – 4294967295
 Default: 5184.
22. **daymsnfeccstm64**
 Day threshold for MS-N-FECC bin forSTM-64.
 Type: Integer.
 Range: 0 – 4294967295
 Default: 497644.
23. **qhmsnfeccstm256**
 15-minute threshold for MS-N-FECC bin forSTM-256.
 Type: Integer.
 Range: 0 – 4294967295
 Default: 20736.
24. **daymsnfeccstm256**
 Day threshold for MS-N-FECC bin forSTM-256.
 Type: Integer.
 Range: 0 – 4294967295
 Default: 1990656.
25. **qhmsses**
 15-minute threshold for MS_N_SES and MS_F_SES bin.
 Type: Integer.
 Range: 0-810
 Default: 10.
26. **daymsses**
 Day threshold for MS_N_SES and MS_F_SES bin.
 Type: Integer.
 Range: 0-77760
 Default: 15.

27. **qhmsuas**
    15-min threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
28. **daymsuas**
    Day threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters - reset profile parameters of type RS-MS

1. **qhrtrrsnbbe**
   15-minute reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **dayrtrrsnbbe**
   Day reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
3. **qhrtrrsnes**
   15-minute reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 20.
4. **dayrtrrsnes**
   Day reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **qhrtrrsnses**
   15-minute reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrrsnses**
   Day reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 0.

7. **`qhrtrrsnuas`**
   15-minute reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **`dayrtrrsnuas`**
   Day reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 0.
9. **`qhrtrmsbbestm1`**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 2400.
10. **`dayrtrmsbbestm1`**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 43200.
11. **`qhrtrmsbbestm4`**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 0-691199100
    Default: 9600.
12. **`dayrtrmsbbestm4`**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 172800.
13. **`qhrtrmsbbestm16`**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-2764799100
    Default: 38400.
14. **`dayrtrmsbbestm16`**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 691200.

15. **qhrtrmsbbestm64**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 153600.
16. **dayrtrmsbbestm64**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 2764800.
17. **qhrtrmsbbestm256**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 614400.
18. **dayrtrmsbbestm256**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 11059200.
19. **qhrtrmses**
    15-minute reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
20. **dayrtrmses**
    Day reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 15.
21. **qhrtrmsnfeccstm64**
    15-minute reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
22. **dayrtrmsnfeccstm64**
    Day reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.

23. **qhrtrmsnfeccstm256**
15-minute reset threshold for MS-N-FECC bin forSTM-256.
Type: Integer.
Range: 0 – 4294967295
Default: 0.

24. **dayrtrmsnfeccstm256**
Day reset threshold for MS-N-FECC bin forSTM-256.
Type: Integer.
Range: 0 – 4294967295
Default: 0.

25. **qhrtrmsses**
15-minute reset threshold for MS_N_SES and MS_F_SES bin.
Type: Integer.
Range: 0-810
Default: 0.

26. **dayrtrmsses**
Day reset threshold for MS_N_SES and MS_F_SES bin.
Type: Integer.
Range: 0-77760
Default: 0.

27. **qhrtrmsuas**
15-min reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-900
Default: 0.

28. **dayrtrmsuas**
Day reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-86400
Default: 0.

Spec block parameters - profile parameters of type ODU

1. **qhodupbbeodu1**
15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-18377100
Default: 32000.

2. **dayodupbbeodu1**
Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-1764201600
Default: 61000.

3. **`qhodupes`**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 50.
4. **`dayodupes`**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P..
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **`qhodupses`**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 10.
6. **`dayodupses`**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 15.
7. **`qhodupuas`**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **`dayodupuas`**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 20.

Spec block parameters - reset profile parameters of type ODU

1. **`qhrtrodupbbeodu1`**
   15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-18377100
   Default: 200.
2. **`dayrtrodupbbeodu1`**
   Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-1764201600
   Default: 4800.

3. **qhrtrodupes**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 5.
4. **dayrtrodupes**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 15.
5. **qhrtrodupses**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrodupses**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 0.
7. **qhrtrodupuas**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **dayrtrodupuas**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 0.

Spec block parameters - profile parameters of type HO-VC

1. **qhvcbbevc3**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
2. **dayvcbbevc3**
   Day threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

3. **qhvcbbevc4**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbevc4**
   Day threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcbbevc44c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
6. **dayvcbbevc44c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
7. **qhvcbbevc416c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
8. **dayvcbbevc416c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
9. **qhvcbbevc464c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
10. **dayvcbbevc464c**
    Day threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
    Type: Integer.
    Range: 0-691113600
    Default: 48000.

11. **qhvcesvc3**
15-minute threshold for VC_N_ES and VC_F_ES VC3 bin.
Type: Integer.
Range: 0-900
Default: 150.
12. **dayvcesvc3**
Day threshold for VC_N_ES and VC_F_ES VC3 bin.
Type: Integer.
Range: 0-86400
Default: 600.
13. **qhvcesvc4**
15-minute threshold for VC_N_ES and VC_F_ES for VC4 bin.
Type: Integer.
Range: 0-900
Default: 180.
14. **dayvcesvc4**
Day threshold for VC_N_ES and VC_F_ES for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 1500.
15. **qhvcesvc44c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-900
Default: 180.
16. **dayvcesvc44c**
Day threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-86400
Default: 1500.
17. **qhvcesvc416c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-900
Default: 180.
18. **dayvcesvc416c**
Day threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-86400
Default: 1500.

19. **qhvcesvc464c**
15-minute threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 180.
20. **dayvcesvc464c**
Day threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 1500.
21. **qhvcsesvc3**
15-minute threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-810
Default: 15.
22. **dayvcsesvc3**
Day threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-77760
Default: 20.
23. **qhvcsesvc4**
15-minute threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-810
Default: 15.
24. **dayvcsesvc4**
Day threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-77760
Default: 20.
25. **qhvcsesvc44c**
15-minute threshold for VC_N_SES and VC_F_SES for VC44C bin.
Type: Integer.
Range: 0-810
Default: 15.
26. **dayvcsesvc44c**
Day threshold for VC_N_SES and VC_F_SES for VC44C bin.
Range: 0-77760
Default: 20.

27. **qhvcsesvc416c**
15-minute threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-810
Default: 15.

28. **dayvcsesvc416c**
Day threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-77760
Default: 20.

29. **qhvcsesvc464c**
15-minute threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-810
Default: 15.

30. **dayvcsesvc464c**
Day threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-77760
Default: 20.

31. **qhvcuasvc3**
15-minute threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-900
Default: 15.

32. **dayvcuasvc3**
Day threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-86400
Default: 20.

33. **qhvcuasvc4**
15-minute threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-900
Default: 15.

34. **dayvcuasvc4**
Day threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 20.

35. **`qhvcuasvc44c`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
36. **`dayvcuasvc44c`**
    Day threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
37. **`qhvcuasvc416c`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
38. **`dayvcuasvc416c`**
    Day threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
39. **`qhvcuasvc464c`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
40. **`dayvcuasvc464c`**
    Day threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
41. **`qhpjcpos`**
    15-minute threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
42. **`daypjcpos`**
    Day threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

43. **`qhpjcneg`**
15-minute threshold for negative PJC bin.
Type: Integer.
Range: 0-1800000
Default: 30.

44. **`daypjcneg`**
Day threshold for negative PJC bin.
Type: Integer.
Range: 0-172800000
Default: 2880.

Spec block parameters - reset profile parameters of type HO-VC

1. **`qhrtrvcbbevc3`**
15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
Type: Integer.
Range: 0-7199100
Default: 200.

2. **`dayrtrvcbbevc3`**
Day reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
Type: Integer.
Range: 0-691113600
Default: 4800.

3. **`qhrtrvcbbevc4`**
15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
Type: Integer.
Range: 0-7199100
Default: 200.

4. **`dayrtrvcbbevc4`**
Day reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
Type: Integer.
Range: 0-691113600
Default: 4800.

5. **`qhrtrvcbbevc44c`**
15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
Type: Integer.
Range: 0-7199100
Default: 200.

6. **`dayrtrvcbbevc44c`**
Day reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
Type: Integer.
Range: 0-691113600
Default: 4800.

7. **qhrtrvcbbevc416c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
8. **dayrtrvcbbevc416c**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
9. **qhrtrvcbbevc464c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
10. **dayrtrvcbbevc464c**
   dayrtrvcbbevc464c
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
11. **qhrtrvcesvc3**
   15-minute reset threshold for VC_N_ES and VC_F_ES VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 10.
12. **dayrtrvcesvc3**
   Day reset threshold for VC_N_ES and VC_F_ES VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 60.
13. **qhrtrvcesvc4**
   15-minute reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
   Type: Integer.
   Range: 0-900
   Default: 20.
14. **dayrtrvcesvc4**
   Day reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
   Type: Integer.
   Range: 0-86400
   Default: 150.

15. **qhrtrvcesvc44c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-900
Default: 20.
16. **dayrtrvcesvc44c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-86400
Default: 150.
17. **qhrtrvcesvc416c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-900
Default: 20.
18. **dayrtrvcesvc416c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-86400
Default: 150.
19. **qhrtrvcesvc464c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 20.
20. **dayrtrvcesvc464c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 150.
21. **qhrtrvcsesvc3**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-810
Default: 0.
22. **dayrtrvcsesvc3**
Day reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-77760
Default: 0.

23. **qhrtrvcsesvc4**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
24. **dayrtrvcsesvc4**
    Day reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
25. **qhrtrvcsesvc44c**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
26. **dayrtrvcsesvc44c**
    Day reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
27. **qhrtrvcsesvc416c**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
28. **dayrtrvcsesvc416c**
    Day reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
29. **qhrtrvcsesvc464c**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
30. **dayrtrvcsesvc464c**
    Day reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.

31. **`qhrtrvcuasvc3`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
32. **`dayrtrvcuasvc3`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
33. **`qhrtrvcuasvc4`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
34. **`dayrtrvcuasvc4`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
35. **`qhrtrvcuasvc44c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
36. **`dayrtrvcuasvc44c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
37. **`qhrtrvcuasvc416c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
38. **`dayrtrvcuasvc416c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

39. **`qhrtrvcuasvc464c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
40. **`dayrtrvcuasvc464c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
41. **`qhrtrpjcpos`**
    15-minute reset threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
42. **`dayrtrpjcpos`**
    Day reset threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.
43. **`qhrtrpjcneg`**
    15-minute reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
44. **`dayrtrpjcneg`**
    Day reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.

Spec block parameters - profile parameters of type PATH

1. **`qhcvpsts1`**
   15-minute threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-57599100
   Default: 15.
2. **`daycvpsts1`**
   Day threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-4294967295
   Default: 125.

3. **`qhcvpsts3c`**
   15-minute threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 0-57599100
   Default: 25.
4. **`daycvpsts3c`**
   Day threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 4294967295
   Default: 250.
5. **`qhcvpsts12c`**
   15-minute threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 0-57599100
   Default: 75.
6. **`daycvpsts12c`**
   Day threshold for threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 4294967295
   Default: 750.
7. **`qhcvpsts48c`**
   15-minute threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 0-57599100
   Default: 300.
8. **`daycvpsts48c`**
   Day threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 4294967295
   Default: 3000.
9. **`qhcvpsts192c`**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-57599100
   Default: 1200.
10. **`daycvpsts192c`**
    Day threshold for CV-P and CV-PFE bin for STS192C.
    Type: Integer.
    Range: 0-4294967295
    Default: 12000.

11. **qhespsts1**
    15-minute threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-900
    Default: 12.
12. **dayespsts1**
    Day threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-86400
    Default: 100.
13. **qhespsts3c**
    15-minute threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-900
    Default: 20.
14. **dayespsts3c**
    Day threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-86400
    Default: 200.
15. **qhespsts12c**
    15-minute threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-900
    Default: 60.
16. **dayespsts12c**
    Day threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-86400
    Default: 600.
17. **qhespsts48c**
    15-minute threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-900
    Default: 240.
18. **dayespsts48c**
    Day threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-86400
    Default: 2400.

19. **`qhespsts192c`**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-900
    Default: 240.
20. **`dayespsts192c`**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-86400
    Default: 9600.
21. **`qhsesp`**
    15-minute threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-810
    Default: 3.
22. **`daysesp`**
    Day threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 7.
23. **`qhuasp`**
    15-minute threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
24. **`dayuasp`**
    Day threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.
25. **`qhppjcgen`**
    15-minute threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
26. **`dayppjcgen`**
    Day threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

27. **qhnpjcgen**
    15-minute threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
28. **daynpjcgen**
    Day threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
29. **qhppjcdet**
    15-minute threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
30. **dayppjcdet**
    Day threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
31. **qhnpjcdet**
    15-minute threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
32. **daynpjcdet**
    Day threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Spec block parameters - profile parameters of type ENET

1. **qhpdege**
   15-minute threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 187500.
2. **daypdege**
   Day threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.

3. **`qhpdevcg`**
   15-minute threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 10.
4. **`daypdevcg`**
   Day threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
5. **`qhedfe1ge`**
   15min EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 1875000.
6. **`dayedfe1ge`**
   24 h EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.
7. **`qhedfc1ge`**
   15min EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 10.
8. **`dayedfc1ge`**
   24 h EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
9. **`qhedfe10ge`**
   15min EDFE for 10GbE Ethernet port.
   Type: Integer.
   Range: 0-2147483647
   Default: 18750000.
10. **`dayedfe10ge`**
    24 h EDFE for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 90000000.

11. **qhedfc10ge**
    15min EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
12. **dayedfc10ge**
    24 h EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.
13. **qhedfe1gevcg**
    15min EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
14. **dayedfe1gevcg**
    24 h EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
15. **qhedfc1gevcg**
    15min EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
16. **dayedfc1gevcg**
    24 h EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
17. **qhedfe10gevcg**
    15min EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
18. **dayedfe10gevcg**
    24 h EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.

19. `qhedfc10gevcg`
15min EDFC for 10GbE VCG port.
Type: Integer.
Range: 0-2147483647
Default: 100.
20. `dayedfc10gevcg`
24 h EDFC for 10GbE VCG port.
Type: Integer.
Range: 0-4294967295
Default: 500.

Spec block parameters - reset profile parameters of type ENET

1. `qhrtrpdege`
15-minute reset threshold for PDE bin for GE.
Type: Integer.
Range: 0-183000000
Default: 937500.
2. `dayrtrpdege`
Day reset threshold for PDE bin for GE.
Type: Integer.
Range: 0-4294967295
Default: 4500000.
3. `qhrtrpdevcg`
15-minute reset threshold for PDE bin for VCG.
Type: Integer.
Range: 0-183000000
Default: 2.
4. `dayrtrpdevcg`
Day reset threshold for PDE bin for VCG.
Type: Integer.
Range: 0-4294967295
Default: 5.
5. `qhrtredfe1ge`
15min reset threshold for EDFE of 1GbE Ethernet port.
Type: Integer.
Range: 0-1371951220
Default: 937500.
6. `dayrtredfe1ge`
24h reset threshold for EDFE of 1GbE Ethernet port.
Type: Integer.
Range: 0-4294967295
Default: 4500000.

7. `qhrtredfc1ge`
   15min reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 2.
8. `dayrtredfc1ge`
   24h reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
9. `qhrtredfe10ge`
   15min reset threshold for EDFE of 10GbE Ethernet port.
   Type: Integer.
   0-2147483647
   Default: 9375000.
10. `dayrtredfe10ge`
    24h reset threshold for EDFE of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 45000000.
11. `qhrtredfc10ge`
    15min reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    0-2147483647
    Default: 20.
12. `dayrtredfc10ge`
    24h reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
13. `qhrtredfe1gevcg`
    15min reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
14. `dayrtredfe1gevcg`
    24h reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.

15. **qhrtredfc1gevcg**
    15min reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
16. **dayrtredfc1gevcg**
    24h reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
17. **qhrtredfe10gevcg**
    15min reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    0-2147483647
    Default: 20.
18. **dayrtredfe10gevcg**
    24h reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
19. **qhrtredfc10gevcg**
    15min reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
20. **dayrtredfc10gevcg**
    24h reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

Spec block parameters- profile parameters of type LOVC

1. **qhvcbbevc12**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 9000.
2. **dayvcbbevc12**
   Day threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 12000.

3. **qhvcbbelovc3**
15minute -threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
Type: Integer.
Range: 0-7199100
Default: 36000.
4. **dayvcbbelovc3**
Day threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
Type: Integer.
Range: 0-691113600
Default: 48000.
5. **qhvcesvc12**
15-minute threshold for VC_N_ES and VC_F_ES for VC12 bin.
Type: Integer.
Range: 0-900
Default: 120.
6. **dayvcesvc12**
Day threshold for VC_N_ES and VC_F_ES for VC12 bin.
Type: Integer.
Range: 0-86400
Default: 350.
7. **qhvceslovc3**
15-minute threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
Type: Integer.
Range: 0-900
Default: 150.
8. **dayvceslovc3**
Day threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
Type: Integer.
Range: 0-86400
Default: 600.
9. **qhvcsesvc12**
15-minute threshold for VC_N_SES and VC_F_SES for VC12 bin.
Type: Integer.
Range: 0-810
Default: 15.
10. **dayvcsesvc12**
Day threshold for VC_N_SES and VC_F_SES for VC12 bin.
Type: Integer.
Range: 0-77760
Default: 20.

11. **qhvcseslovc3**
    15-minute threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
12. **dayvcseslovc3**
    Day threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
13. **qhvcuasvc12**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
14. **dayvcuasvc12**
    Day threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
15. **qhvcuaslovc3**
    15-minute threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
16. **dayvcuaslovc3**
    Day threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters- reset profile parameters of type LOVC

1. **qhrtrvcbbevc12**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 50.
2. **dayrtrvcbbevc12**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 1200.

3. **qhrtrvcbbelovc3**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **dayrtrvcbbelovc3**
   Day reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
5. **qhrtrvcesvc12**
   15-minute reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 5.
6. **dayrtrvcesvc12**
   Day reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 35.
7. **qhrtrvceslovc3**
   15-minute reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 10.
8. **dayrtrvceslovc3**
   Day reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 60.
9. **qhrtrvcsesvc12**
   15-minute reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
10. **dayrtrvcsesvc12**
   Day reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-77760
   Default: 0.

11. **qhrtrvcseslovc3**
    15-minute reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
12. **dayrtrvcseslovc3**
    Day reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
13. **qhrtrvcuasvc12**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
14. **dayrtrvcuasvc12**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
15. **qhrtrvcuaslovc3**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
16. **dayrtrvcuaslovc3**
    Day reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

Spec block parameters.- Profile Parameters of Type LOPATH

1. **qhcvvt1**
   15-minute threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-14399100
   Default: 75.
2. **daycvvt1**
   Day threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-1382313600
   Default: 750.

3. **`qhesvt1`**
   15-minute threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
4. **`dayesvt1`**
   Day threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.
5. **`qhsesvt1`**
   15-minute threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-810
   Default: 40.
6. **`daysesvt1`**
   Day threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-77760
   Default: 900.
7. **`qhuasvt1`**
   15-minute threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
8. **`dayuasvt1`**
   Day threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.

Command parameters - profile parameters of type DS3

1. **`qhcvl`**
   15-minute threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0-7200000
   Default: 387.
2. **`daycvl`**
   Day threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0- 691200000
   Default: 3865.

3. **`qhesl`**
   15-minute threshold for Es-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
4. **`dayesl`**
   Day threshold for ES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
5. **`qhesal`**
   15-minute threshold for ESA-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
6. **`dayesal`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
7. **`qhesbl`**
   15-minute threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
8. **`dayesbl`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
9. **`qhsesl`**
   15-minute threshold for SES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-810
   Default: 4.
10. **`daysesl`**
    Day threshold for SES-L bin of DS3 Line.
    Type: Integer.
    Range: 0-77760
    Default: 40.

11. **qhcvp**
    15-minute threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-7200000
    Default: 382.
12. **daycvp**
    Day threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-691200000
    Default: 3820.
13. **qhesp**
    15-minute threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **dayesp**
    Day threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **qhesap**
    15-minute threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
16. **dayesap**
    Day threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
17. **qhesbp**
    15-minute threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.

18. **dayesbp**
Day threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 250.

19. **qhsesp**
15-minute threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-810
Default: 4.

20. **daysesp**
Day threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-77760
Default: 40.

21. **qhuasp**
15-minute threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-900
Default: 10.

22. **dayuasp**
Day threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 10.

23. **qhsasp**
15-minute threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-900
Default: 2.

24. **daysasp**
Day threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 8.

Command parameters - profile parameters of type DS1

1. **`qhcvsfp`**
   15-minute threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 72.
2. **`daycvsfp`**
   Day threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0- 1048575
   Default: 691.
3. **`qhcvesfp`**
   15-minute threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 13296.
4. **`daycvesfp`**
   Day threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-1048575
   Default: 132960.
5. **`qhesp`**
   15-minute threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 65.
6. **`dayesp`**
   Day threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 648.
7. **`qhsesp`**
   15-minute threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-810
   Default: 10.
8. **`daysesp`**
   Day threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 250.

9. **qhuasp**
   15-minute threshold for UAS-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 10.
10. **dayuasp**
    Day threshold for UAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 10.
11. **qhsasp**
    15-minute threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
12. **daysasp**
    Day threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 17.

Example output

The following example shows the successful completion of a ED-TCA-PROF command by the NE:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ED-TCA-PROF::RS-MS,DTAG: qhrsnes=10,qhrsnbbe=6″
;
```

ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax:

ED-TMAT3:*tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Type: AID
   Range: TMuxAPortAID
   The keyword **all** is not allowed.
   Default: None
3. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **ds3pn** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.
2. **fmt** – Signal Format
   This parameter shows the format for the DS3 signal
   Type: Alphanumeric value set. A hyphen ("-") is allowed in the name additionally.
   Range:
   - C-BIT: C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format. M23 format contains seven DS2 signals, which contain DS1 signals.

   Default: C-BIT.

3. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
   This parameter enables/disables the outgoing DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
4. **`outpm`** – Outgoing DS3 Near End Path PM Enable
   This parameter enables/disables the outgoing DS3 near end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
5. **`pmode`** – Port Mode
   This parameter sets the mode of the port.
   Type: Alphanumeric value set.
   Range:
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

   Default: NMON.
6. **`servcond`** – Service Condition
   This parameter indicates its status concerning service involvement.
   Type: Alphanumeric value set.
   Range:
   - INSERVICE: Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

   Default: INSERVICE.
7. **`tca`** – DS3 PM TCA
   This parameter assigns a DS3 PM TCA profile name to the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   - DEFAULT
   - DEFAULT0

   Default: DEFAULT0.

Output format

In response to a valid RTRV-TMAT3 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

None.

Detailed behavior description

The command modified the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command. The modification are reported by means of an ED-TMAT3 notification for port.

Example input/output

```
ED-TMAT3:UNITE-NE:1-1-#-#-21-tma1:123456:::OUTPM=DISABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-TMAT3:1-1-#-#-21-1::OUTPM=disable:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-TMAT3`** command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Related TL1 messages

- "RTRV-TMAT3: Retrieve asynchronous adjunct port of the TransMUX card" (p. 3-3029)
- "ED-TMAT3 notification" (p. 3-1310)
- "ENT-TMAT3 notification" (p. 3-1941)
- "DLT-TMAT3 notification" (p. 3-263)

References

GR-199-CORE (ED-rr)

ED-TMAT3 notification

Output format

Command syntax: **`ED-TMAT3`**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier of a TransMUX asynchronous adjunct port.
   Type: AID
   Range: TMuxAPortAID
   The keyword **`all`** is allowed up to shelf level.
   Default: None.

Spec block parameters

1. **`ds3pn`** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.
2. **`fmt`** – Signal Format
   This parameter shows the format for the DS3 signal
   Type: Alphanumeric value set. A hyphen ("-") is allowed in the name additionally.
   Range:
   - C-BIT: C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format. M23 format contains seven DS2 signals, which contain DS1 signals.

   Default: C-BIT.
3. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
   This parameter enables/disables the outgoing DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
4. **`outpm`** – Outgoing DS3 Near End Path PM Enable
   This parameter enables/disables the outgoing DS3 near end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.

5. **pmode** – Port Mode
   This parameter sets the mode of the port.
   Type: Alphanumeric value set.
   Range:
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

   Default: NMON.
6. **servcond** – Service Condition
   This parameter indicates its status concerning service involvement.
   Type: Alphanumeric value set.
   Range:
   - INSERVICE: Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

   Default: INSERVICE.
7. **tca** – DS3 PM TCA
   This parameter assigns a DS3 PM TCA profile name to the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   - DEFAULT
   - DEFAULT0

   Default: DEFAULT0.

Example input/output

```
ED-TMAT3:UNITE-NE:1-1-#-#-21-tma1:123456:::OUTPM=DISABLE;

   UNITE-NE 00-11-28 08:00:00
M  123456 COMPLD
;

   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-28:ED-TMAT3:1-1-#-#-21-tma1::outpm=disable:”
;
```

ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack

The command **ED-ULSDCC-L3** is used to set respectively modify values of layer 3 parameter in existing respectively created OSI stacks. Parameter in the area address field of NSAP respectively NET are to set respectively modify. Parameter in the SYSTEM field and SEL field of NSAP respectively NET are predefined. Further layer 3 parameter to set/modify are IS level, IS domain, RA prefix and RA destination node.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-ULSDCC-L3**:*tid*::*ctag*::*osinode*:[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured, see command "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951).
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. **`autoencap`** – Auto Encapsulation
   The parameter *autoencap* is used to enable or disable the Auto Encapsulation function. The parameter is only supported for osinode1. The parameter shall be omitted for all other osinodes.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
2. **`l3afi`** – NSAP/NET Authority and Format Identifier
   This parameter sets or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99
   Default: 39. This value identifies the iso_dcc format.
3. **`l3idi`** – NSAP/NET initial Domain Identifier
   This parameter consists of two values:
   - a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
   - a 1 semi-octet (1-digit hexadecimal) IDI pad part. The parameter is used for set and retrieve values.

   Type: Hexadecimal number.
   Range: 0000 – FFFF
   Default:
   SONET: IDI = 840; IDI pad = F
   SDH: IDI = 000; IDI pad = 0.
4. **`l3dfi`** – NSAP/NET DSP Format Identifier
   This parameter (2-digit hex) NSAP is used to set or retrieve the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 00 – FF.
   Default: 80. This value represents the SONET DSP format.
5. **`l3org`** – NSAP/NET Organization Identifier
   This parameter of NSAP is used to set or retrieve the NSP (allocated Company Code).
   Type: Hexadecimal number.
   Range: 000000 – FFFFFF.
   Default: 000000.
6. **`l3res`** – NSAP/NET Reserved
   This NSAP part currently has not been assigned a specific purpose.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.

7. **`l3rd`** – NSAP/NET Routing Domain
This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.

8. **`l3area`** – NSAP/NET Area Identifier
This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.

9. **`l3isisnl`** – Layer 3 IS Level
This parameter indicates the network router type: ES-IS (level1, value L1), IS-IS (level2, value L2) and partition repair capability (value3).
Note: This parameter is not reported when output parameter *osinode* = scnnode.
Type: Alphanumeric string.
Range:
   - L1: this means level1, IS-IS, only allowed when parameter l3radesnode=0
   - L12: this means level2, IS-IS which contains also level1, ES-IS
   - L12PRC : this means partition repair capable (PRC) and contains also level1 and level2.

Default: L1.

10. **`l3domain`** – Layer3 IS Domain
Per node in a NE, a domain group must be specified. This will be done with this parameter which can be set and retrieved. All nodes in the NE with the same domain-number are interconnected and can communicate without the need for inter-domain routing via RA-PREFIX.
Note: This parameter is not reported when output parameter *osinode* = scnnode.
Type: Alphanumeric string.
Range:
   - 1 for osinode1
   - 1 – 2 for osinode2 to osinode8.

Default: 1.

11. **`l3raprefix`** – Layer3 Reachable Address Prefix
This value specifies an area address or a part of an area address representing an area address group. This parameter is needed when nodes in different routing domains (see *l3domain*) must communicate. The node must forward then CLNP PDUs to the RA-Destination-Node if the destination NSAPs of CLNP PDUs match with the PA-Prefix.
Note: This parameter is not reported when output parameter *osinode* = scnnode.
Type: Hexadecimal number.

Range:

- NONE,
- a string of 1 to 13 bytes: first byte (most significant): 10 – 99; rest: 0000 – FFFF.

The value represents an area address or a part of an area address beginning with AFI.
Default: empty string.

12. **l3radesnode** – Layer3 Reachable Address Destination Node
This parameter is in combination with parameter l3raprefix necessary. The value represents the destination node in the same NE. An internal connection over a domain-boundary is established between this node and the RA-Destination-Node, with IS-IS disabled over that connection.
Note: This parameter is not reported when output parameter *osinode* = scnnode.
Type: AID and digit value.
Range:
    - 0, osinode{2-8} for osinode1
    - 0, osinode1 for osinode2 to osinode 8.

    Value 0 indicates that functionality of RA-Prefix is disabled.
Default: 0.
13. **nsapthlev** – Number of OSI Nodes Threshold Level
This parameter indicates the NSAP count threshold level for monitoring the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the osinode parameter is located. A threshold value of zero (0) indicates that NSAP threshold alarming is disabled. The parameter is omitted for the SCN node.
Type: Numeric string.
Range: 0 no alarming; 1 -250 threshold alarming.
Default: 0

Output format

If the **ED-ULSDCC-L3** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command sets or modifies the parameter of layer 3 in a node as specified in the spec_block of the command line. Result should be an update of the OSI stack.

Modification of any of the parameters *l3domain*, *l3afi*, *l3idi*, *l3dfi*, *l3org*, *l3res*, *l3rd*, *l3area*, *l3isisnl*, *autoencap* leads to a reset of the DCF controller and disconnection of management communication.

The update is reported by means of **ED-ULSDCC-L3** notification.

Example input/output

The following example shows the result of **ED-ULSDCC-L3**:

```
ED-ULSDCC-L3:NE-
      UNITE::4744843::osinode8:l3afi=39,l3idi=840F,l3dfi=80,l3org=100100,l3res=0000,l3rd=0020,l
   NE-UNITE 01-01-15 11:59:02
M  4744843 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-ULSDCC-L3** command.

Validation rule: *osinode* must be in the created state

The *osinode* parameter for which changes should be applied must be in the created state, i.e. the value must be either osinode1 or the node must be successfully created via **ENT-ULSDCC-L3**.

Error code:

`SNVS – Status, Not in Valid State, osinode not in created state`

Validation rule: Address area value – RA-prefix value

If an Address Area value will be changed and an existing RA-Prefix matches with the concerning part of the new Area Address the provisioning of the new NSAP Area Address must be rejected.

Error code:

`IDNV – Input, Data Not Valid, NSAP Area Addres matches with RA-Prefix`

Validation rule: RA-prefix value – Address area value

If a RA-Prefix will be changed respectively initiated and the value matches with the concerning part of the NSAP Area Address of this node the command must be rejected. The prefix should be specific enough, so that it matches not with NSAPs in both domains.

Error code:

`IDNV – Input, Data Not Valid, RA-Prefix matches with the NSAP Area Address`

Validation rule: *l3radesnode* value must be different from osinode

The command will be rejected when parameter *l3radesnode* is equal to the *osinode* parameter of the node.

Error code: `IDNV – Input, Data Not Valid, equal node IDs`

Validation rule: *l3domain* value must be different for current node and *l3radesnode*

If the value for the Domain Group (*l3domain* parameter) is equal for this node and the destination node (*l3radesnode* parameter) the command must be rejected.

Error code: `IDNV – Input, Data Not Valid, equal l3domain`

Validation rule: *l3radesnode* value and *l3isisnl* value

The *l3isisnl* parameter value L1 (Level1) is only allowed when *l3radesnode*=0

Error code: `IDNV – Input, Data Not Valid, constraint l3radesnode`

Validation rule: consistency of *l3radesnode* with *l3raprefix*

If the value for the RA Destination Node (l3radesnode parameter) is equal to 0 while the value of the RA Prefix (l3raprefix parameter) is not NONE the command must be rejected.

Error code:

`IDNV – Input, Data Not Valid, l3prefix and l3radesnode inconsistent`

Validation rule: *l3domain* can not be changed for *osinode1*

If *osinode* = osinode1 *l3domain* changes should be rejected.

Error code:

`IDNV – Input, Data Not Valid, can’t change l3domain for osinode1`

Validation rule: only one single node in a separate domain supported

The command is rejected when *l3domain* has an other value than 1 and there is already an other osinode which has a value other than 1 for *l3domain*.

Error code:

`IDNV – Input, Data Not Valid, only a single node in separate domain supported`

Validation rule: consistency *l3domain* with *l3radesnode* for nodes 2 to 8

The command is rejected when *l3domain* for osinode2 to osinode8 has a value of 1 and *l3radesnode* is not equal to 0.

Error code:
`IDNV – Input, Data Not Valid, inconsistency l3domain with l3radesnode`

Validation rule: *l3radesnode* must be in the created state

If the destination node (*l3radesnode* parameter) is not in the created state, the command must be rejected.

Error code:
`SNVS – Status, Not in Valid State, l3radesnode not in created stat`

Validation rule: Check for IP configuration of LAN1

If it is tried to set *autoencap* to ENABLE without a valid IP address and subnetwork mask configured for LAN port LAN1, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

Validation rule: Check if Auto Encapsulation is still in use by ES64U pack

If it is tried to set *autoencap* to DISABLE while there is an ES64U pack provisioned in the system with a capacity set to 20G, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, Auto Encapsulation still in use`

Validation rule: Check if manual MCN routes exist

If it is tried to set *autoencap* to DISABLE while there are still manual MCN routes provisioned for any of the LAN ports, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, Manual MCN routes still exist`

Related TL1 messages

- “RTRV-ULSDCC-L3: Retrieve layer 3 related parameters” (p. 3-3049)
- “ENT-ULSDCC-L3: Create a further OSI node” (p. 3-1951)
- “DLT-ULSDCC-L3: Delete a node ID other than node-1” (p. 3-265)

ED-ULSDCC-L3 notification

The command **ED-ULSDCC-L3** effects a communication SW-reset (protocol stack) and the management connection is lost. Therefore it cannot be guaranteed that a notification could be returned as described below.

Output format

The **ED-ULSDCC-L3** notification has the following specific syntax:

ED-ULSDCC-L3::*osinode*:*spec_block*

Output parameters

Command parameters

1. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured, see command "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951).
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. **autoencap** – Auto Encapsulation
   The parameter *autoencap* is used to enable or disable the Auto Encapsulation function. The parameter is only supported for osinode1. The parameter shall be omitted for all other osinodes.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
2. **l3afi** – NSAP/NET Authority and Format Identifier
   This parameter sets or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99
   Default: 39. This value identifies the iso_dcc format.
3. **l3idi** – NSAP/NET initial Domain Identifier

This parameter consists of two values:

- a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
- a 1 semi-octet (1-digit hexadecimal) IDI pad part. The parameter is used for set and retrieve values.

Type: Hexadecimal number.
Range: 0000 – FFFF
Default:
SONET: IDI = 840; IDI pad = F
SDH: IDI = 000; IDI pad = 0.

4. **`l3dfi`** – NSAP/NET DSP Format Identifier
This parameter (2-digit hex) NSAP is used to set or retrieve the format for the rest of the address.
Type: Hexadecimal number.
Range: 00 – FF.
Default: 80. This value represents the SONET DSP format.
5. **`l3org`** – NSAP/NET Organization Identifier
This parameter of NSAP is used to set or retrieve the NSP (allocated Company Code).
Type: Hexadecimal number.
Range: 000000 – FFFFFF.
Default: 000000.
6. **`l3res`** – NSAP/NET Reserved
This NSAP part currently has not been assigned a specific purpose.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.
7. **`l3rd`** – NSAP/NET Routing Domain
This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.
8. **`l3area`** – NSAP/NET Area Identifier
This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.

9. **l3isisnl** – Layer 3 IS Level
   This parameter indicates the network router type: ES-IS (level1, value L1), IS-IS (level2, value L2) and partition repair capability (value3).
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Alphanumeric string.
   Range:
   - L1: this means level1, IS-IS, only allowed when parameter l3radesnode=0
   - L12: this means level2, IS-IS which contains also level1, ES-IS
   - L12PRC : this means partition repair capable (PRC) and contains also level1 and level2.

   Default: L1.
10. **l3domain** – Layer3 IS Domain
    Per node in a NE, a domain group must be specified. This will be done with this parameter which can be set and retrieved. All nodes in the NE with the same domain-number are interconnected and can communicate without the need for inter-domain routing via RA-PREFIX.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Alphanumeric string.
    Range:
    - 1 for osinode1
    - 1 – 2 for osinode2 to osinode8.

    Default: 1.
11. **l3raprefix** – Layer3 Reachable Address Prefix
    This value specifies an area address or a part of an area address representing an area address group. This parameter is needed when nodes in different routing domains (see *l3domain*) must communicate. The node must forward then CLNP PDUs to the RA-Destination-Node if the destination NSAPs of CLNP PDUs match with the PA-Prefix.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Hexadecimal number.
    Range:
    - NONE,
    - a string of 1 to 13 bytes: first byte (most significant): 10 – 99; rest: 0000 – FFFF.

    The value represents an area address or a part of an area address beginning with AFI.
    Default: empty string.

12. **l3radesnode** – Layer3 Reachable Address Destination Node
This parameter is in combination with parameter l3raprefix necessary. The value represents the destination node in the same NE. An internal connection over a domain-boundary is established between this node and the RA-Destination-Node, with IS-IS disabled over that connection.
Note: This parameter is not reported when output parameter *osinode* = scnnode.
Type: AID and digit value.
Range:
    - 0, osinode{2-8} for osinode1
    - 0, osinode1 for osinode2 to osinode 8.

    Value 0 indicates that functionality of RA-Prefix is disabled.
Default: 0.
13. **nsapthlev** – Number of OSI Nodes Threshold Level
This parameter indicates the NSAP count threshold level for monitoring the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the *osinode* parameter is located. A threshold value of zero (0) indicates that NSAP threshold alarming is disabled. The parameter is omitted for the SCN node.
Type: Numeric string.
Range: 0 no alarming; 1 -250 threshold alarming.
Default: 0

Example output

```
   NE-UNITE 01-01-13 01:57:42
A  000 REPT DBCHG
   ”TIME=01-47-43,DATE=01-01-13:ED-ULSDCC-L3::osinode8:
   l3afi=39,l3idi=840F,l3dfi=80,l3org=100100,l3res=0000,
   l3rd=0020,l3area=4711,l3isisnl=Ll2,l3domain=2,
   l3raprefix=,l3radesnode=0,nsapthlev=48”
;
```

ED-USER: Edit user

The **ED-USER** command modifies a user's own security parameters.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-USER**:*tid*::*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **screen** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG
     Data Base Change notifications – changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes). The user will receive **REPT DBCHG** messages.
     The value DBCHG shall always be used together with the value STCHG.
   - STCHG
     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **REPT DBCHG** messages.
     The value STCHG shall always be used together with the value DBCHG.
   - PSCHG
     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS
     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.

- ONNS

  ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
- OWN

  Responses to own commands. The user will receive no other notifications.

  Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.
- NA

  Not applicable. This value is reserved for users with uidclass=MEMADMIN. The notification behavior is controlled in this case by the user class.

Default: OWN
If more than one value is set the values will be concatenated by '&'.
Example: DBCHG&STCHG&ALARMS

Output format

After receiving the **ED-USER** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

If the **ED-USER** command does not alter existing user security parameters, the NE provides a normal completion response.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

This command updates the user data of the user *uid* (User ID) who requests the command. The value of *screen* (Message Screening) will be updated according the parameter value.

It will be reported via the corresponding ED-USER notification – see "ED-USER notification" (p. 3-1326) section.

The execution of an ED-USER command is entered in the Security Information Log.

When the **ED-USER** command is successfully executed, the user session is not terminated and is unaffected. It will continue to use the previous values. The changes will take effect the next time the user logs in.

Example input/output

The following example shows the successful completion of the **ED-USER** command by the NE:

ED-USER:LT-UNITE-123::123456:::screen=DBCHG&STCHG;

```
   sid date time
M  ctag DENY
   IIAC
   /* Input, Invalid ACcess identifier, unknown AID */
;
```

Error responses

Validation rule: screen value not allowed for *uid*

A user is not able to add a message type to their *screen* parameter list if:

- the message type was never added a security administrator or
- the message type has been removed by a security administrator

Error code:

`IDNV – Input, Data Not Valid, screen value was not set by security administrator`

Related TL1 messages

"RTRV-USER: Retrieve user" (p. 3-3059)

ED-USER notification

Output format

The **ED-USER** notification has the following specific syntax: **ED-USER**:*uid*::*spec_block*

Output parameters

Command parameters

1. **uid** – User Identifier
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 1 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
   Default: not applicable

Spec block parameters

1. **screen** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG
     Data Base Change notifications – changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes). The user will receive REPT DBCHG and REPT EVT FXFR messages.
   - STCHG
     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **REPT DBCHG** messages.
   - PSCHG
     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS
     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
   - ONNS

ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.

- OWN

  Responses to own commands. The user will receive no other notifications.

  Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.

- NA

  Not applicable. This value is reserved for users with uidclass=MEMADMIN. The notification behavior is controlled in this case by the user class.

Default: OWN
If more than one value is set the values will be concatenated by ’&’.
Example: DBCHG&STCHG&ALARMS

Example output

The example shows a normal notification after an ED-USER command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ED-USER:nbguser1::
   screen=DBCHG&STCHG″
;
```

ED-USER-SECU: Edit user security

The **ED-USER-SECU** command is used by an administrator to edit the security parameters associated with a user.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No.

Input format

Command syntax: **ED-USER-SECU**:*tid*:*uid*:*ctag*::,[*new_pid*],,[*upc*][:*spec_block*];

Input parameters

> **Important!** During a software download (CIT to NE) it may happen that the *Navis*® Optical EMS -NE connection get lost. This is caused by the *tmout* (Inactivity Timeout Period) parameter. A value of 0 will disable the timeout function (see Spec block parameters description below).

Note: The new setting becomes operational only after a new login.

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **uid** – User ID
   ED-USER-SECU input
   Specifies the user ID (*uid*) name which should be edited.
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **new_pid** – New Password ID
   This parameter specifies the new password ID for the addressed *uid*. This is an optional parameter and in the absence of a value the default is the current value. Password strings are transmitted in non-encrypted form in the **ACT-USER** command; are encrypted when stored in the NE; and are never transmitted from the NE.
   Type: character string (case-sensitive)
   Range: 6 to 10 legal characters. Valid passwords consist of at least two

non-alphabetic characters with at least one special character. The special character can be one of the required non-alpha characters. For example, TEST1+ is a valid password. The first character of a password must be a letter.
Default: For the two superusers the default password is: UNITE+01 for User ID LUC01 and UNITE+02 for User ID LUC02. For other users: not applicable.

5. **`upc`** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation “&” is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also “List of user privilege codes” (p. 15-1).

Spec block parameters

1. **`alw_login`** – Allow Login
   This parameter displays the enable/disable state of a User ID.
   A User ID can be automatically disabled by the system when the time specified by parameter *usrage* (see “RTRV-NE-SECU: Retrieve NE security” (p. 3-2552)) expires or via the **`ED-USER-SECU`** command. A User ID can be enabled via the **`ED-USER-SECU`** command.
   If a User ID is disabled, logins with that User ID are denied.
   Type: Alphanumeric value set.
   Range:
   - YES: User ID is enabled
   - NO: User ID is disabled

   Default: YES

2. **`newuid`** – New UID
   This parameter allows to rename the 2 predefined super-users. It cannot be used to rename a non-super-user login. .
   uid the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **`ACT-USER`** command are not allowed. Valid *uid* values for the **`ACT-USER`** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in

the NE.
Type: Alphanumeric string (case-sensitive)
Range: 5 – 10 characters
The *uid* ALL is not allowed in any combination of upper or lower cases.
Default: not applicable

3. **`page`** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90
4. **`screen`** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string. An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG

     Data Base Change notifications – changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes). The user will receive **`REPT DBCHG`** messages.

     The value DBCHG shall always be used together with the value STCHG.
   - STCHG

     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **`REPT DBCHG`** messages.

     The value STCHG shall always be used together with the value DBCHG.
   - PSCHG

     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS

     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
   - ONNS

     ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
   - OWN

     Responses to own commands. The user will receive no other notifications.

     Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.
   - NA

Not applicable. This value is reserved for users with uidclass=MEMADMIN. The notification behavior is controlled in this case by the user class.

Default: OWN

If more than one value is set the values will be concatenated by ’&’.

Example: DBCHG&STCHG&ALARMS

Restriction: When *uidclass* is set to MEMADMIN, screen must be set to NA.

5. **`tmout`** – Inactivity Timeout Period

   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.

   Type: Integer.

   Range: 0 – 999 [minutes]

   Default: 30

6. **`uidclass`** – UID Class

   This is the user class of a *uid*.

   Type: Alphanumeric value set.

   Range:

   - OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
   - MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **`RTRV-AO`**.

   Default: OTHER

Output format

After receiving the **`ED-USER-SECU`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

If the **`ED-USER-SECU`** command does not alter existing user security parameters, the NE provides a normal completion response. An example is when the *upc* of the *uid* is P3 and an ED-USER-SECU command is received which sets the *upc* to P3.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

This command updates the user data of the user specified in the *uid* parameter. The values of *upc* (User Privilege Code List), *alw_login* (Allow Login), *screen* (Message Screening), *page* (Password Aging Interval), *tmout* (Inactivity Timeout Interval) will be updated according the parameter values. *new_pid* updates the *pid* (Password ID). *newuid* updates the uid (User ID).

It will be reported via the corresponding ED-USER-SECU notification – see "ED-USER-SECU notification" (p. 3-1335) section.

The NE always has two preinstalled users with full privileges in all functional categories. These two users will be referred to as super-user. The NE will not allow adding additional super-user type users. It will not be possible to remove either of the original super-user login IDs. An super-users's *upc* cannot be modified using the **ED-USER-SECU** command.

An existing non-super-user login ID (*uid*) cannot be renamed. It must be deleted and re-entered.

A superuser login ID (*uid*) can be renamed by a security administrator (privilege code S5 in category Security Administration).

The **ED-USER-SECU** command enables an Security Administrator to modify the password, user privilege code(s), and/or user security parameters of the non-administrative users. An Security Administrator may modify the password and/or user security parameters of the other Security Administrators including super-users (except the user privilege code).

If the *uid* is currently logged in when the **ED-USER-SECU** command is successfully executed, the user session identified by the *uid* is not terminated and is unaffected. It will continue to use the previous values. The changes will take effect the next time the user logs in.

Example input/output

The following example shows the successful completion of the **ED-USER-SECU** command by the NE:

```
ED-USER-SECU:LT-UNITE-123:nbguser1:123456::,pass13+,,P3&M3;&PM2:screen;=
   DBCHG&STCHG;,page=70,tmout=0,^alw_login=YES;
   sid date time
M  ctag DENY
   ICNV
   /* Input, Command Not Valid, extra AID */
;
```

This example shows how the superuser's *uid* can be changed:

```
ED-USER-SECU:LT-UNITE-123:LUC01:123456::,,,:newuid=Admin1;
```

```
   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Invalid *uid*

If the NE receives an ED-USER-SECU command with an invalid (unknown) *uid* value.

Error code: `IDNV – Input, Data Not Valid, invalid UID`

Validation rule: Invalid *upc*

If the NE receives an ED-USER-SECU command with a *upc* value=P5&M5&T5&S5&PM5 for a non-administrator login account or a upc=S0 or one super-user tries to make the other super-user a non-super-user.

Error code: `IDNV – Input, Data Not Valid, invalid UPC`

Validation rule: newuid has the same value of an existing *uid*

If the NE receives an ED-USER-SECU command from a user trying to change with the *newuid* the *uid* of the addressed user id to an existing *uid* different from the addressed user id.

Error code: `IDNV – Input, Data Not Valid, duplicate UID`

Validation rule: newuid defined for non superuser *uid*

If the NE receives an ED-USER-SECU command from a user trying to change the *uid* of a non super-user.

Error code: `IDNV – Input, Data Not Valid, newuid for non superuser`

Validation rule: disabling of superusers

If the NE receives an ED-USER-SECU command from a user trying to disabling *uid* of a super-user.

Error code: `IDNC – Input, Data Not Consistent`

Validation rule: Compatibility *screen*, *uidclass* parameter

If the network element receives an ED-USER-SECU command with uidclass=MEMADMIN and screen !=NA it shall reject the command with:

Error code:

`IDNV – Input, Data Not Valid, incompatible screen, uidclass parameters`

Related TL1 messages

- "DLT-USER-SECU: Delete user security" (p. 3-272)
- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

ED-USER-SECU notification

Output format

See definition of "REPT DBCHG: Report data base change" (p. 3-2118).

The **ED-USER-SECU** notification has the following specific syntax:

ED-USER-SECU:*uid*:,,[*upc*]:*spec_block*

Output parameters

Command parameters

1. **uid** – User ID
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 1 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
   Default: not applicable
   If the *uid* was changed via the **ENT-USER-SECU** command and a value of *newuid* is different from the old *uid* then the *uid* contains in the notification the value of old *uid*.
2. **upc** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation "&" is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also "List of user privilege codes" (p. 15-1).

Spec block parameters

1. **alw_login** – Allow Login
   This parameter displays the enable/disable state of a User ID.
   A User ID can be automatically disabled by the system when the time specified by parameter *usrage* (see “RTRV-NE-SECU: Retrieve NE security” (p. 3-2552)) expires or via the **ED-USER-SECU** command. A User ID can be enabled via the **ED-USER-SECU** command.
   If a User ID is disabled, logins with that User ID are denied.
   Type: Alphanumeric value set.
   Range:
   - YES: User ID is enabled
   - NO: User ID is disabled

   Default: YES
2. **newuid** – New UID
   This parameter allows to rename the 2 predefined super-users. It cannot be used to rename a non-super-user login. .
   uid the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 5 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
   Default: not applicable
3. **page** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90
4. **screen** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string An ’&’ is also allowed for concatenation of values.
   Range:
   - DBCHG

     Data Base Change notifications – changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes). The user will receive **REPT DBCHG** messages.
   - STCHG

State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **REPT DBCHG** messages.

- PSCHG

  Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
- ALARMS

  Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
- ONNS

  ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
- OWN

  Responses to own commands. The user will receive no other notifications.

  Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.
- NA

  Not applicable. This value is reserved for users with uidclass=MEMADMIN. The notification behavior is controlled in this case by the user class.

Default: OWN
If more than one value is set the values will be concatenated by ’&’.
Example: DBCHG&ALARMS

5. **tmout** – Inactivity Timeout Period
   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.
   Type: Integer.
   Range: 0 – 999 [minutes]
   Default: 30
6. **uidclass** – UID Class
   This is the user class of a *uid*.
   Type: Alphanumeric value set.
   Range:
   - OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
   - MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **RTRV-AO**.

   Default: OTHER

Example output

The example shows a normal notification after an ED-USER-SECU command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:ED-USER-SECU:nbguser1:,,
   P3&M3&PM2:screen=DBCHG&STCHG,page=70,tmout=0,
   newuid=nbguser1,
   alw_login=YES”
;
```

This is an example notification in case of the user id was changed for a super-user:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:ED-USER:nbguser1::
   SCREEN=DBCHG&STCHG”
;
```

ED-VC3: Edit VC3 tributary

The command **ED-VC3** is used to provision all parameters related to the tributary VC3.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: no

Input format

Command syntax: **ED-VC3**:*tid:aid:ctag:::spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   AID of a SDH tributary. The keyword **all** is allowed on slot, port and tributary level. The so called ″dynamic parameters″ is reported, if a **all** is used at the tributary level.

   The ″dynamic parameters″ are:

   - *ptrc*, *ptrcst*, *siglb*, *trailstate*.
     Type: AID
     Range: STS1NumAID.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
4. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
5. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
6. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
7. **`hptimmon`** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

8. **`loca`** – Ring A-node TID
   This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
   loca is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in "RTRV-HDR: Retrieve header" (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
9. **`locz`** – Ring Z-node TID
   This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
   locz is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
10. **`neppm`** – Near-End Path PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
11. **`pjepm`** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).

12. **`pmhpfeses`** – Performance Monitoring Path Far-End SES Threshold
    This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.
13. **`pmhpneses`** – Performance Monitoring Path Near-End SES Threshold
    This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.
14. **`ptca`** – Path TCA Profile Pointer
    Type: Alphanumeric string.
    Range: string with up to 24 characters.
    Default: DEFAULT0.
15. **`ptrcrfmt`** – Path Trace Read Format (J1)
    The read format of J1, the incoming path trace (ptrc).
    Type: numeric string.
    Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string.
    - 64: for 64 byte string.
      Default: 2.
16. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in BURST or POISSON mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON
17. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.
    Two thresholding methods are applicable:
    - TR-only (also known as the implicit reset method)
    - TR-RTR (also known as the explicit reset method).
      Type: Alphanumeric value set.
      Range: TR, TRRTR (default).

18. **tmonmd** – Tributary Monitoring Mode
    This parameter indicates the mode of the tributary.
    Type: Alphanumeric value set.
    Range:
    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.
19. **tribsdhpn** – Tributary Alarm Severity Assignment Profile
    This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: string with a maximum of 24 characters.
    Default: DEFAULT.

Output format

If the ED-VC3 command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command **ED-VC3** is also applicable to VC4 tributaries. Even though a VC4 signal is not a concatenated VC3 signal in a strict sense, from a pointer processor viewpoint it can be considered to be comprised of three VC3 signals which have the properties of a concatenated signal. Thus, it is not necessary to have a separate ED-VC4 command. Similarly, ED-VC3 is also applicable to VC4-4C and other ″concatenated″ tributaries.

Even if the command is used for a concatenated signal, it can be applied to any of the tributaries (for example, to the second tributary of the concatenated signal). However, only the parameters pertaining to the first tributary of a concatenated signal are used by the software to determine system operation. If the signal were changed from a concatenated signal to non-concatenated, then the parameters for each of the tributaries would be used by the software.

The command allows to modify all attributes listed above as input spec-block parameters. The modification is reported by means of an ED-VC3 notification for tributary.

Change of J1 expected format - FM impact

When the expected J1 format (ptrcrfmt) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J1 (exphptrc) and the currently received J1 (ptrc).

When the expected J1 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Signal degrade/signal fail mode provisioning

If the attribute *sdsfmode* is changed then for all tributaries belonging to the same STS-12/VC4-4c group the attribute will also be changed according to value specified in the ED request.

A STS-12/VC4-4c group is composed of all tributaries whose Simple_STS1NumAID is in the range from (k*12) +1 to (k+1)*12 with k=0..63. The maximal value of k is derived from the port rate.

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port and the Simple_STS1NumAID for that STS-12/VC4-4c group;

i.e. the Simple_STS1NumAID field in the STS1NumAID is set to <M>&&-<N>

- with M is first STS-1/AU-3 number of the STS-12/VC4-4c group,
- with N is last STS-1/AU-3 number of the STS-12/VC4-4c group.

If additional parameters besides *sdsfmode* are changed within the ED request then it is also valid to split the autonomous reporting in up to 3 notifications:

- one group notification for all tributaries with smaller Simple_STS1NumAID as the one which is specified in the ED request.
- one single notification for the tributary which is specified in the ED request.
- one group notification for all tributaries with greater Simple_STS1NumAID as the one which is specified in the ED request.

Disabling of PM monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PM counting
2. all current bins related to the PM monitor is deleted
3. all history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Disabling of PJE Monitor

The disabling of a pointer justification monitor shall have the following impacts w.r.t. existing PM registers:

1. stop PJE counting
2. all current bins related to the PJE monitor is deleted
3. all history bins related to the PJE monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of *pm enable attributes*.

PM monitors can be disabled explicitly by means of the *pm enable* attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts (ID 52924) the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Example input/output

```
ED-VC3:UNITE-NE:1-1-#-#-3-1-1:123456:::tmonmd=MON,tribsdhpn=Default,degthr=-
      6,dexcthr=-3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30;

   UNITE-NE 00-10-14 14:59:09
M  123456 COMPLD
;

   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ”TIME=14-59-09,DATE=00-10-14:ED-VC3:1-1-#-#-3-1-1:: tmonmd=MON,tribsdhpn=Default,
degthr=-6,dexcthr=-
      3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30:”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the ED-VC3 command.

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:
`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`

Since several monitor points can be enabled/disabled within one command, the **ED** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **ED** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Not more than one PJE PM monitor per port

Check that no PJE PM monitor is already enabled for any tributary of the port to which the addressed tributary belongs to.

Error code:
`SSRE – Status, System Resources Exceeded, limit of PJE counters exceeded`

Validation rule: Check CCB against *istd* of related port

Check wether the CCB fits to istd of the related port

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`

Validation rule (ONNS specific): *loca*, *locz* changes only allowed when *portClass* = EDGE and part of a BLSR/MS-SPRING group

Setting the *loca*, *locz* parameters away from the default value is only allowed when the related *portClass* = EDGE and the port is part of a BLSR/MS-SPRING protection group.

Error code:
`IDNV – Input, Data Not Valid, related portClass not EDGE or port not in BLSR/MS-SPRING group`

Validation rule (ONNS specific): *loca*, *locz* changes only allowed when tributary not involved in a cross-connection

Setting the *loca*, *locz* parameters are inhibited when the tributary is already input or output tributary of a cross-connection.

Error code:
`IDNV – Input, Data Not Valid, tributary already part of a cross-connection`

Validation rule (ONNS specific): Maximum number of tributaries with *loca*, *locz* entries must not exceed 100

It is checked that the maximum number of triburaries with loca, locz parameters set (i.e. having a value different from the default value) does not exceed 100.

Error code:
`SSRE – Status, System Resources Exceeded, limit of tribs with loca, locz exceeded`

Validation rule (ONNS specific): *hptimmon* changes to ENABLE only allowed when corresponding port is not of *portClass* INNI

Setting the *hptimmon* parameter to ENABLE is only allowed when the port the VC-3 is located on does not have the port class INNI.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Related TL1 messages

“RTRV-STS1: Retrieve STS1 tributary” (p. 3-2946)

References

GR-199-CORE (ED-*rr*)

ED-VC3 notification

Output format

Syntax: **`ED-VC3`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   AID of a SDH tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when sdsfmode=POISSON.
   Type: numeric value set.:
   Range (exponents of 10): -5, -6 (default), -7, -8, -9.
4. **`crslpbkrate`** – Cross-connect loopback Rate
   It is independent from the current interface standard and therefore both SDH and SONET values are allowed.
   The parameter *crslpbkrate* is not reported if the parameter *crlspbkstat* indicates that no loopback is set.
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C

- STS1
- STS3
- STS12
- STS48
- STS192

5. **`crslpbkstat`** – Cross-connect loopback status
   State of cross- connect loopback.
   Type: Alphanumeric value set.
   Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
6. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
7. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
8. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`hptimmon`** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).

10. **loca** – Ring A-node TID
    This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
    loca is introduced specifically for use related to ONNS application.
    Type: TID or empty-string.
    Range: Please refer to the TID definition in "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: "". This is displayed as \"\" in output direction.
11. **locz** – Ring Z-node TID
    This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
    locz is introduced specifically for use related to ONNS application.
    Type: TID or empty-string.
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: "". This is displayed as \"\" in output direction.
12. **neppm** – Near-End Path PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
13. **pjepm** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
14. **pmhpfeses** – Performance Monitoring Path Far-End SES Threshold
    This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.

15. **pmhpneses** – Performance Monitoring Path Near-End SES Threshold
This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
16. **ptca** – Path TCA Profile Pointer
Type: Alphanumeric string.
Range: string with up to 24 characters.
Default: DEFAULT0.
17. **ptrcrfmt** – Path Trace Read Format (J1)
The read format of J1, the incoming path trace (ptrc).
Type: numeric string.
Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string.
    - 64: for 64 byte string.
    Default: 2.
18. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in BURST or POISSON mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON
19. **substr** – Substructured
This parameter indicates if the tributary is substructured.
Substructure for LO cross-connections:
A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.
The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).
A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **ED-SUBS** command.
Substructure for conversion cross-connections:
A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.

The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.
A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.
Type: Alphanumeric value set.

Range:

- YES: Automatically sub structured for LO cross-connections
- MANSUB: Manually sub structured for LO cross-connections
- CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
- NO (default): Not sub structured, not used for conversion cross-connections

20. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (also known as the implicit reset method)
    - TR-RTR (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR (default).
21. **`tmonmd`** – Tributary Monitoring Mode
This parameter indicates the mode of the tributary.
Type: Alphanumeric value set.
Range:
    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.
22. **`tribsdhpn`** – Tributary Alarm Severity Assignment Profile
This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: string with a maximum of 24 characters.
Default: DEFAULT.

Example output

```
ED-VC3:UNITE-NE:1-1-#-#-3-1-1:123456:::tmonmd=MON,tribsdhpn=Default,degthr=-
      6,dexcthr=-3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30;

   UNITE-NE 00-10-14 14:59:09
M  123456 COMPLD
;

   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ”TIME=14-59-09,DATE=00-10-14:ED-VC3:1-1-#-#-3-1-1:: tmonmd=MON,tribsdhpn=Default,
```

```
degthr=-6,dexcthr=-
      3,ptrcrfmt=1,ptca=Default0,neppm=ENABLE,pmhpneses=30,pmhpfeses=30:"
;
```

ED-VCG: Edit virtual concatenation group

The **ED-VCG** command modifies the properties of a virtual concatenation group.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **ED-VCG***[-modifier]*:*tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
The modifier specifies the type of the virtual concatenation group of the unit. Implicit, it indicates the transmission rate for which VCG related data is retrieved. The parameter modifier must be given if a virtual concatenation group on a GE10PL1 or ES64U pack is addressed. Otherwise this parameter has to be omitted.
Type: Alphanumeric value set.
Range: 1GEPL representing a 1 GE port on the GE10PL1 pack; 10GEPL representing a 10 GE port on the GE10PL1 pack, ES64U representing a VCG port at an ES64U subsystem.
2. **tid** – Target Identifier
See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
This parameter specifies the virtual concatenation group (VCG) to be operated on.
Type: AID.
Range: VCGAID as defined in AID structure. The values
   - vall for Simple_VCGAID
   - usall for Simple_UnivSlotAID
   - all for Simple_UnivSlotAID
   - all for Simple_ShelfAID and
   - all for BayAID

   is excluded.
Default: For changing the *eportControl* parameter the 'ALL' qualifier for the port aid is allowed.
4. **ctag** – Correlation Tag
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameter

1. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other eport classes in future.
   This parameter is supported by ports of *eportClass* UNI.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: IPv4 format: Four dot separated numeric string
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
   Restriction: For *modifier* = ES64U, the parameter cannot be modified.
2. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer WAN port or a network WAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER, for a customer WAN port. NETWORK, for a network WAN port.
   Default: NETWORK.
3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *dftpri* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (that means, from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging

mode.
This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *dftvlan* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- 1, 2, 3, … 4092, 4093.
- UNAVAILABLE (default).

5. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
This parameter points to an ASAP profile name of type ENET for the designated AID.
Ethernet port and VCG port have the same ASAP profile named ENET.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: 1 … 24 characters length.
Default: DEFAULT.

6. **`enettca`** – Ethernet TCA Profile Name
This parameter specifies the name of the threshold crossing alert (TCA) profile used for the VCG.
This parameter is not supported if the input parameter *modifier* = ES64U.
If the parameter *enettca* is not supported, then the parameter is omitted.
Type: Alphanumeric string.
Range: 1 … 24 characters length.
Default: DEFAULT0.

7. **`eportClass`** – Ethernet Port Class
This identifies the Ethernet port class of the AID. *eportClass* is used to specify Ethernet ports for use by NN application.
For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
Type: Alphanumeric value set.

Range: Depends on the *modifier*:

- if *modifier* is missing or is 1GEPL, then the value range is: TRADITIONAL, EDGE, UNI
- if *modifier* is 10GEPL, then the value range is: TRADITIONAL, EDGE
- if *modifier* is ES64U, then the value range is: TRADITIONAL

Default: Depends on system parameters Cross Connect Setup (*xcsetup*) and Port Class Default (*portclassdefault*):

- if *xcsetup* is TRADITIONAL, then the default value is TRADITIONAL
- if *xcsetup* is SWISS, then the value of *portclassdefault* is used as default value.
- if *modifier* is ES64U, then the default value is TRADITIONAL

Restriction: For *modifier* = ES64U the parameter cannot be modified.

8. **`eportControl`** – Ethernet Port Control (Ethernet Port Administrative Mode)
   This identifies the administrative mode of Ethernet ports of *eportlass* = EDGE or UNI. For Ethernet ports of other Ethernet port class this parameter is not relevant. *eportControl* is introduced specifically for use by NN application.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: Alphanumeric value set.
   Range: NORMAL (default), MAINT, LOCKOUT .
   Restriction: For *modifier* = ES64U the parameter cannot be modified.
9. **`htmr`** – Hold-Off Time
   The parameter *htmr* specifies the hold off time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it goes to state faulty then this tributary is taken out of transmission by LCAS after the hold off time. This time is configured on group level, i.e. all tributaries have the same value for hold-off time. This parameter only affects NE behavior if lcasmd=ENABLE.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: numeric value set.
   Range: 0 (default), 100, 200, …, 9900, 10000 [milliseconds].
   Restriction: For *modifier* = ES64U the parameter cannot be modified.
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter specifies to which VCG tributaries an Ethernet signal will be mapped. This parameter can only be used (see also error responses) if: the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, and the current value of the VCG parameter *istd* = SONET, and the current value of the VCG parameter *vcat* = ENABLE.
    Restriction: For *modifier* = ES64U the parameter cannot be modified.
    Type: numeric string of the form NX
    - N specifies the count of container type X
    - X specifies the container type (or envelope).

    Range:
    - (N=7, X=3) or (N=21, X=1) if modifier = = 1GEPL
    - (N=64, X=3) or (N=192, X=1) if modifier = 10GEPL.

    Other inputsig parameter values can be set implicitly by changing the parameters *istd* and/or *vcat*.
11. **`istd`** – Interface Standard
    This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
    For *modifier* = ES64U, *modifier* = ES64U or *modifier* = ES64U the parameter is visible with default values for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.

Range:

- SDH
- SONET.

Default: *modifier*: 1GEPL, 10GEPL, or ES64U: The value is derived from the circuit pack parameter *istd*. no *modifier*: As derived from parameter *istddflt* specified at configuration management, section **RTRV-EQPT**.
Restriction: The parameter istd is only allowed if modifier is omitted.

12. **lcasmd** – LCAS Mode
    This parameter specifies the operation mode of the Link Capacity Adjustment Scheme for this VCG. This value is bidirectional.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.
    Range:
    - DISABLE
    - ENABLE

    Default:
    - *modifier* = ES64U: DISABLE
    - other or no *modifier*: ENABLE
13. **localLPID** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *eportClass* UNI.
    The parameter *localLPID* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: numeric value set.
    Range: 0 to 4294967295.
    Default: 0.

14. **localLPIDVCAT** – Local Logical Port Identity on VCAT layer
    The *localLPIDVCAT* provides a unique logical identification of a single VCG port with an *eportClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port, the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like, for example, *networkTNAAddrVCAT*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *localLPIDVCAT* is used by ONNS for switching network connections on VCAT layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: numeric value set.
    Range: 0 to 4294967295.
    Default: 0.
15. **networkTNAType** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *localLPIDVCAT* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.
    Range: IPV4, IPV6, or NSAP
    Default: NSAP
    Restriction: For *modifier* = ES64U, this parameter is not editable.
16. **networkTNATypeVCAT** – Network TNA Type on VCAT level
    The *networkTNATypeVCAT* identifies the type of the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be

evaluated until *eportClass* is set to a value supporting it.
The parameter *networkTNATypeVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of eportClass UNI.
This parameter is supported by ports of eportClass EDGE.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: Alphanumeric value set.
Range: IPV4, IPV6, or NSAP
Default: NSAP
Restriction: For *modifier* = ES64U, this parameter is not editable.

17. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given NetworkTNAType. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
The parameter *networkTNAAddress* is used by ONNS for switching network connections on SONET/SDH layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type:
networkTNAType = IPV4: Four dot separated numeric string (represents 4 Bytes)
networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
networkTNATyp e= NSAP: Hexadecimal number (represents 20 Bytes)
Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Examples:
IPV4: 129.12.3.178
IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000
Restriction: For *modifier* = ES64U, this parameter is not editable.

18. **`networkTNAAddrVCAT`** – Network TNA Address on VCAT level
The *networkTNAAddrVCAT* identifies the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNATypeVCAT*. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
The parameter *networkTNAAddrVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type:
networkTNAType = IPV4: Four dot separated numeric string (represents 4 bytes)
networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 bytes)
networkTNAType = NSAP: Hexadecimal number (represents 20 Bytes)
Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000
Restriction: For *modifier* = ES64U, this parameter is not editable.

19. **`plcrthr`** – Threshold for Partial Loss of Capacity (Receive)
The parameter *plcrthr* specifies the acceptable limit of valid VCG members in receive direction (refer to the *outmbs* parameter of all related VCGTRIB entities) for which the PLCR alarm will not yet be raised. The PLCR alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plcrthr* parameter – 1)] (if *plcrthr* ≥ 2). The PLCR alarm shall never be raised if *plcrthr* is set to 1.
The PLCR alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plcrthr* parameter (or if the number of those VCG members is 0 – in this case the TLCR alarm will be raised instead). This

parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plcrthr* is not supported, then the parameter is omitted.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 1, 2, 3, …, 192.
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192
ES64U	n × 1	n

Restriction: For *modifier* = ES64U, this parameter is not editable.

20. **`plctthr`** – Threshold for Partial Loss of Capacity (Transmit)
The parameter *plctthr* specifies the acceptable limit of valid VCG members in transmit direction (refer to the *inmbs* parameter of all related VCGTRIB entities) for which the PLCT alarm will not yet be raised. The PLCT alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plctthr* parameter – 1)] (if *plctthr* ≥ 2). The PLCR alarm shall never be raised if *plctthr* is set to 1.
The PLCT alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plctthr* parameter (or if the number of those VCG members is 0 – in this case the TLCT alarm will be raised instead).
This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plctthr* is not supported, then the parameter is omitted.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 1, 2, 3, …, 192.
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192
ES64U	n × 1	n

Restriction: For *modifier* = ES64U, this parameter is not editable.

21. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U.
    Type: Alphanumeric value set.
    Range: DISABLE, STP is disabled, that means, the port is not used to build the spanning tree. ENABLE (default), STP is enabled, that means, the port is used to build the spanning tree.
    As VCGs are by default network ports the default value is to enable STP.
22. **`stpcost`** – Spanning Tree Path Cost
    The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
    If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U.
    If the parameter *stpcost* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 to 200000000.
    Default: 20000.
    The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.
23. **`stpp`** – Spanning Tree Port Priority
    The parameter *stpp* specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database. This parameter is not supported if the input parameter: *modifier* = ES64U.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *stpp* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.
    Default: 128.

24. **`tcamd`** – TCA mode

    This parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable:

    - TR-only (also known as the implicit reset method)
    - TR-RTR (also known as the explicit reset method)

    This parameter is not supported if the input parameter: *modifier* = ES64U.
    If the parameter *tcamd* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: TR, TRRTR.
    Default: TRRTR in SDH mode; TR in SONET mode.

25. **`vcat`** – Virtual Concatenation

    This parameter vcat specifies whether the Ethernet signal is mapped to a number of virtually concatenated tributaries (value = ENABLE) or into a contiguously concatenated tributaries (value = DISABLE). In the latter case, LCAS related parameters become insignificant (values set to default, values cannot be changed). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *vcat* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.

    Range:

    - For *modifier* ES64U: ENABLE (default)
    - otherwise: DISABLE.
    - For *modifier* = ES64U the parameter *vcat* is visible for retrieval, but cannot be modified once the VCG is established.

26. **`vmode`** – VCG Monitoring Mode

    The parameter *vmode* specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.

    Range:

    - MON, for monitored
    - NMON (default), for not monitored.

27. **`vpm`** – VCG Performance Monitoring

    The parameter *vpm* specifies whether the performance monitoring is enabled or disabled for this port.
    This parameter is not supported if the input parameter: *modifier* = ES64U.
    If the parameter *vpm* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.

    Range:

    - DISABLE (default)
    - ENABLE.

28. **`vvidls`** – Valid VLAN Identifier List (static)
    This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using ’&’ and ’&&’ . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (*cnprole* = NETWORK).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *vvidls* is not supported, then the parameter is omitted.
    Type: Alphanumeric string and the character ’&’.
    Range:
    - Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
    - In case the list has no members (empty list) –1.

    Default: –1.
29. **`wtr`** – Wait To Restore Time
    The parameter *wtr* specifies the wait to restore time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it recovers from a faulty state then this tributary is taken in again to transmission by LCAS after the wait-to-restore time. This time is configured on group level, i.e. all tributaries have the same value for the wait-to-restore time. This parameter only affects NE behavior if *lcasmd* = ENABLED.
    For *modifier* = ES64U the parameter is visible with default values for retrieval, but cannot be modified.
    Type: Alphanumeric value set.
    Range: 0, 1, 2, …, 59, 60 [minutes].
    Default: 0.
    Restriction: For *modifier* = ES64U, this parameter is not editable.

Output format

If the NE fully complies with the **`ED-VCG`** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **ED-VCG** shall generate a **REPT DBCHG** message.

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command.

The modification is reported by means of an **ED-VCG** notification.

Impact on *eportControl* when changing *eportClass*

When *eportClass* is changed, *eportControl* is set to NORMAL unless a different value is explicitly specified in the command changing the *eportClass*.

Impact on port parameter settings while *eportClass* is TRADITIONAL

When eportClass has the value TRADITIONAL for an Ethernet port, setting of the following parameter is accepted, but shall have no impact until eportClass is changed away from TRADITIONAL: *eportControl*

Generic ONNS specific behavior - Impact on ONNS related port parameters when not supported by actual *portClass*

Specific behaviors or validations apply for changing or setting ONNS parameters of a port with respect to the actual value of the parameter *portClass*. These is verified first.

In any other case, when an ONNS related port parameter is not supported by the actual *portClass* value of a port, setting of this parameters for this port is accepted, but shall have no impact until *eportClass* is changed to a value where this parameter is supported.

At the moment the *eportClass* is changed to a value where this parameter is supported, the current value of this parameter is considered.

Impact on EDGE related port parameter settings, when *eportClass* is set to EDGE

When the *eportClass* is changed to EDGE, the following parameters of the EDGE parameter set are set to their default values unless all of these parameters are specified in the same command with values different from the default values:

- *networkTNAAddress*
- *localLPID*

Impact on VCAT layer parameter settings, when *eportClass* is set to EDGE

When the *eportClass* is changed to EDGE, the following VCAT layer related parameters are set to their default values:

- *networkTNATypeVCAT*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Impact on EDGE related port parameter settings, if *eportClass* is changed away from EDGE

When *eportClass* is changed away from EDGE, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Impact on UNI related port parameter settings, when *eportClass* is set to UNI

When the *eportClass* is changed to UNI, the following parameters of the UNI parameter set is set to their default values, unless different values are specified in the command setting the *eportClass* to UNI:

- *networkTNATypeVCAT*

The following parameters of the UNI parameter set are mandatory and is set different to their default values:

- *networkTNAAddressVCAT*
- *clientNodeIDVCAT*
- *localLPIDVCAT*

This is covered by corresponding validation rules.

Impact on port parameter settings, when *eportClass* is changed to UNI

When *eportClass* is changed to UNI, the following parameters are set to their default value:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Impact on UNI related port parameter settings, if *eportClass* is changed away from UNI

When *eportClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNAType*
- *networkTNAAddress*

- *clientNodeID*
- *localLPID*

Impact on VCAT layer related port parameter settings, when *eportClass* is changed away from UNI

When *eportClass* is changed away from UNI, the following parameters are set to their default values:

- *networkTNATypeVCAT*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Behavior due to interface standard change

A change of istd is only possible if the VCG port has no cross-connections. A change of *istd* is not dependent from pending alarms or active performance monitoring for the individual tributaries.

A change of *istd* shall set default values for the parameters:

- cnprole, dftpri, dftvlan, enettca, htmr, lcasmd, stdis, stpcost, stpp, tcamd, vmode, vpm, wtr.

A change of istd shall set the parameter vvidls to default (empty list), if the GE pack works not in backward compatible mode (*vlantagmd* = BCMD). If the GE pack works in backward compatible mode (*vlantagmd* = BCMD), then the parameter *vvidls* is not reset to an empty list (VRTSW does not exists, so the *vvidls* is set indirectly by EPORT configuration).

Update affected entities

A change of the value of the parameter lcasmd shall trigger the report of the corresponding tributary changes via a ED-VCGTRIB notification.

A change of the value of the parameter istd shall trigger the deletion of all tributaries of this VCG port and the creation of tributaries for this VCG port with the default values according the new value of *istd*. The deletion is reported by a DLT-VCGTRIB notification. The creation is reported by an ENT-VCGTRIB notification. The deletion of the objects leads to a clearance of pending alarms for these objects. The generation of objects with a different istd generates new alarms for these objects if the alarm condition is still present.

Switching of interface standard - ignore other parameters

If additional parameters beside istd are specified with the ED-rr command they will be ignored by the NE. This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Example for *istd* change from SONET to SDH

```
ED-VCG:LT-UNITE:1-1-#-#-2-V3:654321:::istd=SDH;
```

```
   LT-UNITE 00-11-11 11:11:00
M  654321 COMPLD
;
```

```
   LT-UNITE 00-11-11 11:11:05
A  001 REPT DBCHG
   "TIME=11-11-02,DATE=00-11-11:ED-VCG:1-1-#-#-2-V3::istd=SDH"
;
```

```
   LT-UNITE 00-11-11 11:11:06
A  003 REPT DBCHG
   "TIME=11-11-02,DATE=00-11-11:DLT-VCGTRIB:1-1-#-#-2-V3-1&&-21"
;
```

```
   LT-UNITE 00-11-11 11:11:08
A  005 REPT DBCHG
   "TIME=11-11-05,DATE=00-11-11:ENT-VCGTRIB-VC:1-1-#-#-2-V3-1&-4&-7&-10&-13&-16&-
      19::brstintvl=7,brstthr=30,degthr=-6,dexcthr=-
      3,hptimais=ENABLE,exphptrc=89000000000000000000000000000000,hptimmon=DISABLE,inmbs=DIS
;
```

Implicit Enabling/Disabling of GVRP when Enabling/Disabling STP for a port

When enabling/disabling STP for a network port via parameter stdis GVRP is implicitly enabled/disabled for this port (if enabled via parameter gvrpmode of the virtual switch).

This requirement is only applicable if the input parameter *modifier* = 1GE.

Group notifications for VCG tributaries

For efficiency reasons the notifications for VCG tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the used mapping of the associated vcg port, i.e. the Simple_VCGNumAID field in the VCGNumAID is set to

- 1&&-<N> for non-sequential range
- 1&-4 .. &-<N> for non-sequential range

with N is maximal legal VCG tributary number.

Example:

The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping) 1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping)

Disabling of PM monitor

Two different scenario are relevant w.r.t. disabling of PM monitors in the context of this command:

1. set the ’pm enable’ attribute to disable – affects VCG PM monitors
2. change interface standard – affects VCGTRIB PM monitors due to deletion.

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. Stop PM counting
2. All current bins related to the PM monitor is deleted
3. All history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of ’pm enable attributes’. PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion. Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected .

Applicability of requirements in this section

Following behavior is only applicable for *modifier* = 1GEPL or *modifier* = 10GEPL:

Note that for the OPT2G5 pack, the modifier ″TRANS″ is applicable. However, due to the different parameter set, this has been documented in a separate chapter starting with ID 68300.

Exclusive parameter in a ED-VCG command

If the spec_block of the **`ED-VCG`** command contains the parameter inputsig then no other spec_block parameter is accepted at the same time. Multiple parameter sets within the spec_block including the inputsig parameter is rejected.

Conditions for a parameter change

The system shall only accept value changes of the parameter inputsig if:

- none of the related VCGTRIB entities has been cross-connected,
- all of the related VCGTRIB entities have input membership (inmbs) or output membership (outmbs) disabled,
- if the current value of the parameter *istd* = SONET, and
- if the current value of the parameter *vcat* = ENABLE.

Restricted set of inputsig parameter values

Although the value range of the parameter inputsig may represent more values, the system shall only accept modifications in cases:

- modifier = 1GEPL: Changes to the values 73 and 211 are allowed,
- modifier = 10GEPL: Changes to the values 643 and 1921 are allowed.

Automatic adaptation of the parameter *unequiposig*

If the parameter inputsig has been successfully modified then the parameters unequiposig is set to the same value automatically. The modification of the parameters inputsig and unequiposig (implicitly) is notified by a single ED-VCG notification.

No Automatic creation/deletion of VCGTRIB entities

If the parameter inputsig has been successfully modified then the related VCGTRIB entities is not created or deleted. There shall neither a DLT-VCGTRIB notification nor an ENT-VCGTRIB notification be reported. This implies that there is always the maximum number of related VCGTRIB entities existing. However, some VCGTRIB entities may be insignificant. Modifications of those entities will be stored in the database but have no effect to the transmission signals.

Clear and probable raise of related VCGTRIB alarms

If the parameter inputsig has been successfully modified then all alarms of the related VCGTRIB entities are cleared. New alarms for still significant VCGTRIB entities are generated if the alarm condition is still present for the changed transmission rate.

Exclusive parameter in a ED-VCG command

If the spec_block of the **`ED-VCG`** command contains the parameter vcat then no other spec_block parameter is accepted at the same time. Multiple parameter sets within the spec_block including the vcat parameter is rejected.

Conditions for a parameter change

The system shall only accept value changes of the parameter vcat if: the input parameter *modifier* = 10GEPL, and none of the related VCGTRIB entities has been cross-connected.

Automatic set of default values

If the parameter vcat has been successfully set to DISABLE, then the related default values for the following parameters are set: *htmr*, *lcasmd*, *plcrthr*, *plctthr*, *wtr*.

The system shall not allow to change these parameters, while vcat is set to DISABLE.

Automatic set of values for parameter *inputsig* and *unequiposig*

If the parameter vcat has been successfully modified then the parameters inputsig and unequiposig is set as follows:

- *vcat* = ENABLE: 644 (if the current value of the parameter istd = SDH
- *vcat* = ENABLE: 1921 (if the current value of the parameter istd = SONET)
- *vcat* = DISABLE: 1C64 (if the current value of the parameter istd = SDH)
- *vcat* = DISABLE: 1192 (if the current value of the parameter istd = SONET).

Notification of Automatic Changes

The automatic update of implicitly changed parameters are notified by a single ED-VCG notification.

No Automatic creation/deletion of VCGTRIB entities

If the parameter vcat has been successfully modified then the related VCGTRIB entities is not created or deleted although their significance will change. There shall neither a DLT-VCGTRIB notification nor an ENT-VCGTRIB notification be reported. This implies that there is always the maximum number of related VCGTRIB entities existing. Even if the parameter vcat represents a mapping into contiguous concatenated tributaries, the configuration for those will be controlled by a VCGTRIB entity (name misleading). However, some VCGTRIB entities may be insignificant. Modifications of those entities will be stored in the database but have no effect to the transmission signals. Details about he significance can be obtained in the introduction of the chapter Ethernet Port, in particular in ID 71843.

Clear and probable raise of related VCGTRIB alarms

If the parameter vcat has been successfully modified then all alarms of the related VCGTRIB entities are cleared. New alarms for still significant VCGTRIB entities are generated if the alarm condition is still present for the changed transmission rate.

Automatic set of parameter values for VCGTRIB entities

If the parameter vcat has been successfully modified from the value ENABLE to DISABLE or DISABLE to ENABLE, then the following parameters of all related VCGTRIB entities are set to default values: *inmbs*, *outmbs*. The automatic update of the implicitly changed parameters inmbs and outmbs as well as the consequently changing status parameters inmbst and outmbst is notified by ED-VCGTRIB notifications. All other parameters of VCGTRIB entities keep their values although they may become insignificant.

No side effects for VCGTRIB entities

If the parameter lcasmd has been successfully modified then the related VCGTRIB entities is not influenced although some of their parameters may become insignificant. In this case, a modification of those parameters will be stored in the database but will have no effect on the transmission signal. The motivation is to keep LCAS settings while LCAS can be disabled e.g. for test purposes.

Changing the *vpm* parameter

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

1. Stop PM counting
2. All current bins related to the PM monitor is deleted
3. All history bins related to the PM monitor is deleted.

The number of PM monitors is determined by means of the number of ’pm enable attributes’. PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion. Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected.

Specific behavior of the ES64U pack

The **ENT-VCG** command generates a **REPT DBCHG** message for the creation of the VCG together with a **REPT DBCHG** message for the creation of the VCG tributaries (**ENT-VCGTRIB** notification).

The command establishes a virtual concatenation group together with the virtual concatenation group tributaries belonging to it according to the parameters given with this command. The virtual concatenation group tributary number range is from 1 up to the number of tributaries given by the parameter *inputsig*.

The modification is reported by means of an **ENT-VCG** notification and grouped notifications for VCG tributaries.

Group notification for VCG tributaries

For efficiency reasons the notifications for VCG tributaries shall be sent out as a single notification (so-called group notification) using a range over the AID which depends on the used mapping of the associated VCG port, i.e. the *Simple_VCGNumAID* field in the *VCGNumAID* is set to

- 1&&-***<N>*** for sequential range
- 1&-4…&-***<N>*** for non-sequential range

where ***<N>*** is the maximal legal VCG tributary number.

Example:

The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping)

1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping)

Example input/output

```
ED-VCG:LT-UNITE:1-1-#-#-1-V1:123456:::vmode=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-VCG** command.

Validation rule: command only allowed for VCG on GE*n* or ES64U circuit packs

If the command is not applied to a VCG on a GE*n* or ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: *modifier* mandatory for the GE10PL1 and ES64U circuit pack

The system shall verify that the modifier parameter is provided by the command if the related slot has been successfully provisioned with a GE10PL1 and ES64U circuit pack.

Error code:

```
IDNC – Input, Data Not Consistent, modifier is required for this
circuit pack type
```

Validation rule: *modifier* consistent with given port-rate

The system shall verify that a given modifier parameter is consistent with the rate of the port (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-VCG** command for applicable combinations.

Error code:

```
IDNC – Input, Data Not Consistent, modifier is inconsistent with
existing port rate
```

Validation rule: crossconnection - interface standard change

Check that the interface standard is not changed if there is still a cross-connection applied to any tributary of the whole VCG port. Whole port means all tributaries of a port independent of there configured membership.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections are still existing`

Validation rule: no loopbacks for interface standard change

Check that the interface standard parameter (istd) is not changed when cross-connection loopback is still on this port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: no test access session for *istd* change

Check that *istd* is not changed when a tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code: `SNVS – Status, Not in Valid State, test access session exists`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: ’ALL’ qualifier used for wrong parameters

The ALL qualifier for the port aid is only allowed for changing the *portControl* parameter.

Error code:

`IDNC – Input, Data Not Consistent, ’ALL’ qualifier for port aid not allowed`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:

`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded.`

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **ED** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

ONNS specific validation rule - *eportClass* only settable when *xcsetup* = SWISS

Modification of the *eportClass* parameter is only allowed when xcsetup = SWISS.

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule (ONNS specific): No interface standard change when in use by ONNS

For a 1GEPL or 10GEPL VCG, a modification of the interface standard parameter (*istd*) is not allowed, if this VCG is used by ONNS, i.e if its *eportClass* is different from TRADITIONAL.

Error code: `SNVS – Status, Not in Valid State, SwiSS access possible`

Validation rule: No configuration of unsupported VCG tributary structure for EDGE or UNI 1GEPL VCG

If *eportClass* of a 1GEPL VCG is EDGE or UNI, it is not allowed to configure a tributary structure that is not supported for *eportClass*.

For the tributary structures only the following values of *inputsig* are supported:

- SDH: *inputsig* = 74
- SONET: *inputsig* = 211

Changes to other values shall lead to the following error code.

Error code:

`SNVS – Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`

Validation rule: No configuration of unsupported VCG tributary structure for EDGE 10GEPL VCG

If *eportClass* of a 10GEPL VCG is EDGE, it is not allowed to configure a tributary structure, that is not supported for this *eportClass*.

Allowed combinations of parameters *inputsig* and *vcat* are:

- SDH: *inputsig* = 644 and *vcat* = ENABLE
- SONET: *inputsig* = 643 and *vcat* = ENABLE

Changes to other combinations shall lead to the following error code.

Error code:

`SNVS – Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`

Validation rule (ONNS specific): Unique setting of *localLPID*

It is only allowed to set the parameter *localLPID* to a value different from its default value if this value is unique within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): Unique setting of *localLPID* wrt. *localLinkID*

It is only allowed to set the parameter *localLPID* to a value different from its default value if this value is not already assigned to parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): *localLPID* already assigned to VCG port as *localLPIDVCAT*

It is only allowed to set the parameter *localLPID* to a value different from its default value if this value is not already assigned to parameter *localLPIDVCAT* of any VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): *localLPIDVCAT* already assigned to other VCG port

It is only allowed to set the parameter *localLPIDVCAT* to a value different from its default value if this value is not already assigned to the parameter *localLPIDVCAT* of any other VCG port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): *localLPIDVCAT* already assigned to other EDGE or SONET/SDH UNI port

It is only allowed to set the parameter *localLPIDVCAT* to a value different from its default value if this value is not already assigned to the parameter *localLPID* of any other port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): Unique setting of *localLPIDVCAT* wrt. *localLinkID*

It is only allowed to set the parameter *localLPIDVCAT* to a value different from its default value if this value is not already assigned to the parameter *localLinkID* of any ENNI port within the system.

Error code: `IDNV – Input, Data Not Valid, local LPID value must be unique`

Validation rule (ONNS specific): Check of TNA parameters

It is not possible to set/configure the *networkTNAType* and *networkTNAAddress* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule (ONNS specific): Check of VCAT layer TNA parameters

It is not possible to set/configure the *networkTNATypeVCAT* and *networkTNAAddrVCAT* to not matching value pairs.

Error code:

`IDNV – Input, Data Not Valid, TNA Type and TNA Address do not match`

Validation rule (ONNS specific): Modify of TNA parameters

It is not possible to modify any of the TNA related parameters, if there still exist any cross-connections for the VCG, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule (ONNS specific): Modify of TNA parameters on VCAT layer

It is not possible to modify any of the VCAT layer related TNA related parameters, if there still exist any cross-connections for the VCG, which are of class NN and are assigned to ONNS connections associated to end-to-end network connections.

The parameters are:

- *networkTNATypeVCAT*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`SNVS – Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing`

Validation rule (ONNS specific): Modify of TNA parameters

It is possible to modify the TNA related parameters only

1. if the *eportControl* state is LOCKOUT
2. during the transition of the *eportClass* to a value supporting these parameters

The parameters are:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule (ONNS specific): Modify of TNA parameters on VCAT layer

It is possible to modify the VCAT layer related TNA related parameters only

1. if the *eportControl* state is LOCKOUT
2. during the transition of the *eportClass* to a value supporting these parameters

The parameters are:

- *networkTNATypeVCAT*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`SNVS – Status, Not in Valid State, only allowed for EDGE port in lockout state`

Validation rule (ONNS specific): Setting of *eportClass* to EDGE only possible, if no test access session actve

It is not allowed to change eportClass of a VCG port to EDGE when a VCG tributary related to this port is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Setting of eportClass of a 1GEPL VCG to EDGE only possible, if the configured tributary structure is supported

It is not allowed to change *eportClass* of a 1GEPL VCG to EDGE if the existing tributary structure (configured via *inputsig* parameter) for the corr. VCG is not supported for *eportClass* EDGE and no supported tributary structure is given together with setting the *eportClass* to EDGE.

For EDGE 1GEPL VCGs only the following tributary structures are supported

- SDH: 74
- SONET: 211

Changes to other values shall lead to the following error code.

Error code:

`SNVS – Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`

Validation rule: Setting of eportClass of a 10GEPL VCG to EDGE only possible, if the configured tributary structure is supported

It is not allowed to change *eportClass* of a 10GEPL VCG to EDGE if the existing tributary structure (configured via *istd*, *inputsig*, and *vcat* parameters) for the corr. VCG is not supported for *eportClass* EDGE.

For EDGE 10GEPL VCGs only the following combinations are supported:

- SDH - 644 - ENABLE
- SONET - 643 - ENABLE

Error code:

`SNVS – Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`

Validation rule (ONNS specific): Setting of *eportClass* from EDGE to TRADITIONAL only possible, if the VCG tributaries on the port are not in use by ONNS

For setting *eportClass* from EDGE to TRADITIONAL the following must be fulfilled:

1. No cross-connections with *xconcls* NN must exist on any VCG tributary of the VCG.
2. *eportControl* must be LOCKOUT

Error code: `SNVS – Status, Not in Valid State, VCG is still in use`

Validation rule (ONNS specific): Change the *eportClass* away from EDGE

It is not possible to change the *eportClass* away from EDGE, if *eportControl* is NORMAL. First *eportControl* must be set to LOCKOUT

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule (ONNS specific): Change the *portClass* away from EDGE

It is not possible to change the *eportClass* away from EDGE, if the target of *eportClass* is different to TRADITIONAL.

Error code:

`SNVS – Status, Not in Valid State, Port Class must be TRADITIONAL`

Validation rule (ONNS specific): Complete and Consistent Setting of TNA Parameter required when configured to EDGE VCGs

When assigning TNA information different from the default values to an EDGE VCG, then a complete set for the following parameters must be given:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

For consistency reasons, additionally the values for *networkTNAAddress* and *localLPID* must be different to their default values.

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule (ONNS specific): Consistent Parameters setting required when removing TNA information from EDGE VCGs

When removing TNA information from an EDGE VCG, the following parameters must all be set to their default values:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, Consistent TNA parameter set required`

Validation rule (ONNS specific): No Support of VCAT related TNA Parameters for EDGE VCGs

While the *eportClass* of a VCG is EDGE or during *eportClass* transition towards EDGE, it is not allowed to set any of the following parameters different to their default values:

- *networkTNATypeVCAT*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`IDNV – Input, Data Not Valid, No support of VCAT TNA parameter set for EDGE VCGs`

Validation rule: (ONNS specific): TNA parameters on SDH/SONET layer already assigned on VCAT layer

It is not possible to set non-default TNA parameters on SDH/SONET layer for an EDGE VCG if the desired value of *networkTNAAddress* is already assigned as *networkTNAAddressVCAT* to any VCG in the system.

Error code: `IDNV – Input, Data Not Valid, given TNA already exists`

Validation rule: TNA parameter disjointness between EDGE and UNI

It is not possible to set non-default TNA parameters on SDH/SONET layer for an EDGE VCG, if the desired value of networkTNAAddress is already assigned to any UNI port in the system.

Error code:

`IDNV – Input, Data Not Valid, TNA Address already assigned to UNI port`

This validation means, that UNI and EDGE ports shall not be mixed under one TNA.

Validation rule (ONNS specific): Change the *eportClass* to UNI

It is not possible to change *eportClass* to UNI, if cross-connections on that port are still existing.

Error code:

`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule (ONNS specific): Change the *eportClass* to UNI

It is not possible to change *eportClass* to UNI, if a tributary related to this port is used in a test access session. Either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule (ONNS specific): Change the *eportClass* to UNI

It is not possible to change *eportClass* to UNI, if a cross-connect loopback exists for any tributary related to this port.

Error code:

`SNVS – Status, Not in Valid State, a cross connect loopback exists`

Validation rule (ONNS specific): Change the *eportClass* to UNI

It is not possible to change *eportClass* to UNI, if UNI-N node is not enabled for the NE.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: TNA parameters on VCAT layer already assinged to other VCG port

It is not possible to change TNA parameters on VCAT layer (*networkTNAAddrVCAT* and *networkTNATypeVCAT*) for a VCG port, if already a TNA with identical values is assigned to another VCG port on VCAT layer.

Error code: `SNVS – Status, Not in Valid State, given TNA already exists`

Validation rule: TNA parameters on VCAT layer already assigned to other UNI or EDGE port

It is not possible to change TNA parameters on VCAT layer for a UNI VCG port if already a TNA with identical values (Network TNA Type and Network TNA Address) assigned to another UNI or EDGE port on SDH/SONET layer exists in the system.

Error code: `SNVS – Status, Not in Valid State, given TNA already exists`

Validation rule: Setting of *eportClass* of a 1GEPL VCG to UNI only possible, if the configured tributary structure is supported

It is not allowed to change *eportClass* of a 1GEPL VCG to UNI, if the tributary structure configured via *istd* and *inputsig* parameters for the corr. VCG is not supported for *eportClass* UNI and no supported tributary structure is given together with setting the *eportClass* to UNI.

For UNI 1GEPL VCGs only the following tributary structures are supported:

- SDH – 74
- SONET – 211.

Error code:

`SNVS – Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`

Validation rule: Change the *eportClass* to UNI

It is not possible to change *eportClass* to UNI if UNI-N node is not enabled for the network element.

Error code: `SNVS – Status, Not in Valid State, UNI not enabled`

Validation rule: Check of UNI parameters

It is not possible to change *eportClass* to UNI unless values different from the default values are given for the following UNI parameters

- *clientNodeID*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`IDNV – Input, Data Not Valid, UNI parameters must be different to their DEFAULT values`

Validation rule: Change the *eportClass* away from UNI

It is not possible to change *eportClass* to UNI if *eportControl* is NORMAL. First *eportControl* must be set to LOCKOUT.

Error code:

`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Change the *eportClass* away from UNI

It is not possible to change the *eportClass* away from UNI, if the target of *eportClass* is different to TRADITIONAL.

Error code:
`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

Validation rule: Change the *eportClass* away from UNI

It is not possible to change the *eportClass* away from UNI, if there still exist any cross-connection on any tributary of the port.

Error code: `SNVS – Status, Not in Valid State, Port is still in use`

Validation rule: Modify of UNI parameters

It is not possible to modify UNI related parameters, if the UNI port is not in *eportControl* state LOCKOUT.

The parameters are:

- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, Port must be in LOCKOUT mode`

Validation rule: Modify of UNI parameters

It is not possible to modify UNI related parameters, if there still exist any cross-connection for the UNI port.

The parameters are:

- *clientNodeID*

Error code:
`SNVS – Status, Not in Valid State, Cross-connections still existing`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:
`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to change the *eportClass* to UNI, unless valid values different from the default values are given for the following UNI parameters:

- *clientNodeID*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`IDNV – UNI parameters must be different to their DEFAULT values`

Validation rule: Check of UNI parameters

It is not possible to change the *eportClass* to UNI, unless valid values for the following UNI parameters are given: *clientNodeID*, *networkTNAType*, *networkTNAAddress* and *localLPID*.

Error code:

`IDNV – Input, Data Not Valid, Valid values for UNI parameters required`

Validation rule: Check of UNI parameters

It is not possible to set/configure the *clientNodeID* to the same value of the *uniNetworkNodeID*.

Error code:

`IDNV – Input, Data Not Valid, Client Node ID must differ from Network Node ID`

Validation rule: Check of UNI parameters

It is not possible to set the following UNI parameters to their default values, if *eportClass* is equal to UNI or during setting *eportClass* to UNI:

- *clientNodeID*
- *networkTNAAddrVCAT*
- *localLPIDVCAT*

Error code:

`IDNV – Input, Data Not Valid, UNI port parameters must be different to their DEFAULT values`

Validation rule: Check of SONET/SDH layer parameters

It is not possible to set the following SONET/SDH layer related parameters different to their default values, when *eportClass* is equal to UNI or during setting *eportClass* to UNI:

- *networkTNAType*
- *networkTNAAddress*
- *localLPID*

Error code:

`IDNV – Input, Data Not Valid, No support of TNA parameter set for UNI VCGs`

Validation rule: TNA parameters on VCAT layer already assigned on SDH/SONET layer

It is not possible to set TNA parameters on VCAT layer for a UNI VCG, if the desired value of *networkTNAAddressVCAT* is already assigned as *networkTNAAddress* to any port in the system.

Error code: `SNVS – Status, Not in Valid State, given TNA already exists`

Validation rule: Uniqueness of TNA parameters on VCAT layer per UNI VCG

It is not possible to set TNA parameters on VCAT layer for a UNI VCG, if the desired value of *networkTNAAddressVCAT* is already assigned to a different UNI VCG in the system.

Error code: `SNVS – Status, Not in Valid State, given TNA already exists`

Applicability of requirements in this section

Following behavior is only applicable if the parameter *modifier* has been omitted:

Validation rule: Command only allowed for VCG on Gigabit Ethernet Packs

If the command is not applied to a VCG on a Gigabit Ethernet pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Parameter modification not allowed in backward compatible mode

No modification of the following parameters are allowed if the VLAN tagging mode *vlantagmd* of the pack, to which the specified VCG port belongs to, is set to backward compatible mode (BCMD): *cnprole*, *stdis*, *dftpri*, *dftvlan*, *stpp*, *stpcost*, *rstpmd*, *vvidls*.

Error code:

```
SNVS – Status, Not in Valid State, Modification not allowed in Backward
Compatible Mode
```

Validation rule: Default VLAN ID is already assigned for this system

The default VLAN ID of a Customer port cannot have the same value as the default VLAN Id of another Customer port in the system or any other instantiated VLAN in the system, except for the value 4094.

Error code:

```
IDNC – Input, Data Not Consistent, Default VLAN ID is already assigned
for this System
```

Applicability of this section

The following behavior is only applicable if the input parameter *modifier* is

- 1GEPL or
- 10GEPL

Note that for the OPT2G5 pack, the modifier “TRANS” is applicable.

Validation rule: No other spec_block parameters

If the parameter *inputsig* is modified then any attempts to change other spec_block parameters within the same command shall lead to a rejection of the whole command:

Error code:

`IDNV – Input, Data Not Valid, parameter inputsig must be set individually`

Validation rule: Not allowed for SDH

If the parameter *inputsig* is modified and the current value of the parameter *istd* = SDH, then the command is rejected:

Error code:

`SNVS – Status, Not in Valid State, Not allowed for SDH interface standard`

Validation rule: Not allowed if virtual concatenation mode disabled

If the parameter *inputsig* is modified and the current value of the parameter *vcat* = DISABLE, then the command is rejected:

Error code:

`SNVS – Status, Not in Valid State, Not allowed if VCAT disabled`

Validation rule: Not allowed if a member is cross-connected

If the parameter *inputsig* is modified and at least one related VCGTRIB is cross-connected, then the command is rejected:

Error code:

`SNVS – Status, Not in Valid State, cross-connections are still existing`

Validation rule: Not allowed if a member is looped

If the parameter *inputsig* is modified and at least one related VCGTRIB is looped-back, then the command is rejected:

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: Not allowed if involved in test access

If the parameter *inputsig* is modified and at least one related VCGTRIB is involved in a test access session, then the command is rejected:

Error code:

`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Not allowed if a member is enabled

If the parameter *inputsig* is modified and the input membership or output membership of at least one related VCGTRIB is enabled, then the command is rejected:

Error code:

`SNVS – Status, Not in Valid State, VCG members are still enabled`

Validation rule: Improper Value

If the parameter *inputsig* is modified but the value does not match the defined subset, then the command is rejected:

Error code:

`IDRG – Input, Data Range Error, Value not supported for this modifier and istd`

Validation rule: Ethernet Port Class EDGE not allowed for 10 GE ports

The system shall not allow to set the Ethernet Port Class (parameter eportClass) to EDGE for a 10 GE port (modifier = 10GEPL).

Error code:

`IDNV – Input, Data Not Valid, Port Class not allowed for 10 GE ports`

Validation rule: Ethernet Port Class UNI not allowed for 10 GE ports

The system shall not allow to set the Ethernet Port Class (parameter eportClass) to UNI for a 10 GE port (modifier = 10GEPL).

Error code:

`IDNV – Input, Data Not Valid, Port Class not allowed for 10 GE ports`

Validation rule: Not allowed to change istd for VCG ports on a GE10PL1 pack

The system does not allow to modify the parameter *istd* for VCG ports on a GE10PL1 pack.

Error code:

`IDNV – Input, Data Not Valid, Interface Standard cannot be modified`

The GE10PL1 circuit pack allows only one common interface standard setting for all VCGs on the pack. The interface standard can be changed via the circuit pack parameter istd. A change of the circuit pack interface standard shall update all existing VCG ports on the GE10PL1 pack.

Validation rule: No other spec_block parameters

If the parameter vcat is modified then any attempts to change other spec_block parameters within the same command shall lead to a rejection of the whole command:

Error code:

`IDNV – Input, Data Not Valid, parameter vcat must be set individually`

Validation rule: Not allowed for 1 GE interfaces

If the parameter vcat is modified and the input parameter modifier = 1GEPL, then the command is rejected:

Error code:
`SNVS – Status, Not in Valid State, Not allowed for 1 GE interfaces`

Validation rule: Not allowed if a member is cross-connected

If the parameter vcat is modified and at least one related VCGTRIB is cross-connected, then the command is rejected:

Error code:
`SNVS – Status, Not in Valid State, cross-connections are still existing`

Validation rule: Not allowed if a member is looped

If the parameter vcat is modified and at least one related VCGTRIB is looped-back, then the command is rejected:

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: Not allowed if involved in test access

If the parameter vcat is modified and at least one related VCGTRIB is involved in a test access session, then he command is rejected:

Error code:
`SNVS – Status, Not in Valid State, a test access session exists`

Validation rule: Parameter modification restrictions while vcat is set to DISABLE

While the parameter vcat is set to DISABLE the system shall not allow to change the parameters: *htmr*, *lcasmd*, *plcrthr*, *plctthr*, *wtr*

Error code:
`SNVS – Status, Not in Valid State, Virtual concatenation is disabled`

Related TL1 Messages

"RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)

ED-VCG notification

Output format

The **ED-VCG** notification has the following specific syntax:

ED-VCG*[-modifier]:aid::spec_block;*

Output parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the virtual concatenation group of the unit. Implicit, for the 1GEPL, 10GEPL application it indicates the transmission rate for which VCG related data will be notified. For the ES64U application the transmission rate is defined by the user with the **ENT-VCG** command and input parameter *inputsig*.
   The parameter *modifier* must be given if a virtual concatenation group on a GE10PL1 or an ES64U pack shall be notified. Otherwise this parameter has to be omitted.
   Type: Alphanumeric value set.
   Range: 1GEPL representing a 1 GE port on the GE10PL1 pack; 10GEPL representing a 10 GE port on the GE10PL1 pack, ES64U representing a VCG port at an ES64U subsystem.
2. **aid** – Access Identifier
   This parameter specifies the virtual concatenation group (VCG) to be operated on.
   Type: AID.

   Range: VCGAID as defined in AID structure. The values

   - *vall* for Simple_VCGAID and
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   are excluded.

Spec block parameters

1. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other eport classes in future.

This parameter is supported by ports of *eportClass* UNI.
For *modifier* = ES64U the parameter is visible with default values for retrieval, but cannot be modified.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

2. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer WAN port or a network WAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER, for a customer WAN port. NETWORK, for a network WAN port.
   Default: NETWORK.
3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *dftpri* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (that means, from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U. If the parameter *dftvlan* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - 1, 2, 3, … 4092, 4093.
   - UNAVAILABLE (default).

5. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Ethernet port and VCG port have the same ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
6. **`enettca`** – Ethernet TCA Profile Name
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for the VCG.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   If the parameter *enettca* is not supported, then the parameter is omitted.
   Type: Alphanumeric string.
   Range: 1 … 24 characters length.
   Default: DEFAULT0.
7. **`eportClass`** – Ethernet Port Class
   This identifies the Ethernet port class of the AID. *eportClass* is used to specify Ethernet ports for use by NN application.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: Alphanumeric value set.
   Range: Depends on the *modifier*:
   - if *modifier* is missing or is 1GEPL, then the value range is: TRADITIONAL, EDGE, UNI
   - if *modifier* is 10GEPL, then the value range is: TRADITIONAL, EDGE
   - if *modifier* is ES64U, then the value range is: TRADITIONAL

   Default: Depends on system parameters Cross Connect Setup (*xcsetup*) and Port Class Default (*portclassdefault*):
   - if *xcsetup* is TRADITIONAL, then the default value is TRADITIONAL
   - if *xcsetup* is SWISS, then the value of *portclassdefault* is used as default value.
   - if *modifier* is ES64U, then the default value is TRADITIONAL

   Restriction: For *modifier* = ES64U the parameter cannot be modified.
8. **`eportControl`** – Ethernet Port Control (Ethernet Port Administrative Mode)
   This identifies the administrative mode of Ethernet ports of *eportlass* = EDGE or UNI. For Ethernet ports of other Ethernet port class this parameter is not relevant. *eportControl* is introduced specifically for use by NN application.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: Alphanumeric value set.
   Range: NORMAL (default), MAINT, LOCKOUT .
   Restriction: For *modifier* = ES64U the parameter cannot be modified.

9. **`htmr`** – Hold-Off Time
   The parameter *htmr* specifies the hold off time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it goes to state faulty then this tributary is taken out of transmission by LCAS after the hold off time. This time is configured on group level, i.e. all tributaries have the same value for hold-off time. This parameter only affects NE behavior if lcasmd=ENABLE.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: numeric value set.
   Range: 0 (default), 100, 200, ..., 9900, 10000 [milliseconds].
   Restriction: For *modifier* = ES64U the parameter cannot be modified.
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure depends on interface standard specified for this VCG by *istd* . For definition of *istd* refer to the "Output parameters" section in "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078).
    If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the possible tributary structures depends on interface standard specified for this VCG by istd and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to the "Output parameters" section in "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)..
    Type: numeric string of the form ***NX***
    - ***N*** specifies the count of container type ***X***
    - ***X*** specifies the container type (or envelope).

    Range:

Values	Modifier	Interface standard	Virtual concatenation	Represented signal structure
N = 7, ***X*** = 4	omitted	SDH	not applicable	7 × VC4
N = 7, ***X*** = 4	1GEPL	SDH	ENABLE	7 × VC4
N = 64, ***X*** = 4	10GEPL	SDH	ENABLE	64 × VC4
N = 1, ***X*** = c64	10GEPL	SDH	DISABLE	1 × VC4-64C
N = 21, ***X*** = 1	omitted	SONET	not applicable	21 × STS-1
N = 21, ***X*** = 1	1GEPL	SONET	ENABLE	21 × STS-1
N = 7, ***X*** = 3	1GEPL	SONET	ENABLE	7 × STS-3C
N = 192, ***X*** = 1	10GEPL	SONET	ENABLE	192 × STS-1
N = 64, ***X*** = 3	10GEPL	SONET	ENABLE	64 × STS-3C
N = 1, ***X*** = 192	10GEPL	SONET	DISABLE	1 × STS-192C

Range: Depends on the *modifier*:

- if *modifier* is ES64U, then the value combination of ″N″ and ″X″ is according the following rules:
  istd=SONET
  X = 1 (for STS-1)
  N $\in$ {1, 2, … 192}
- if *modifier* is ES64U, then the parameter *inputsig* is visible for retrieval, but cannot be modified once the VCG is established.

Default:

- modifier = omitted, Interface Standard=SDH, Default Value = 74
- modifier = 1GEPL, Interface Standard=SDH, Default Value = 74
- modifier = 10GEPL, Interface Standard=SDH, Default Value = 644
- modifier = omitted, Interface Standard=SONET, Default Value = 211
- modifier = 1GEPL, Interface Standard=SONET, Default Value = 211
- modifier = 10GEPL, Interface Standard=SONET, Default Value = 1921

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, i.e. VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception:
For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.
For *modifier* = ES64U this parameter is always defined by the **ENT-VCG** command.

11. **istd** – Interface Standard
    This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
    For *modifier* = ES64U, *modifier* = ES64U or *modifier* = ES64U the parameter is visible with default values for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.
    Range:
    - SDH
    - SONET.

    Default: *modifier*: 1GEPL, 10GEPL, or ES64U: The value is derived from the circuit pack parameter *istd*. no *modifier*: As derived from parameter *istddflt* specified at configuration management, section **RTRV-EQPT**.
    Restriction: The parameter istd is only allowed if modifier is omitted.
12. **lcasmd** – LCAS Mode
    This parameter specifies the operation mode of the Link Capacity Adjustment Scheme for this VCG. This value is bidirectional.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.

Range:

- DISABLE
- ENABLE

Default:

- *modifier* = ES64U: DISABLE
- other or no *modifier*: ENABLE

13. **localLPID** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view.
Depending on the *eportClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *eportClass* UNI.
The parameter *localLPID* is used by ONNS for switching network connections on SONET/SDH layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 0 to 4294967295.
Default: 0.

14. **localLPIDVCAT** – Local Logical Port Identity on VCAT layer
The *localLPIDVCAT* provides a unique logical identification of a single VCG port with an *eportClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *eportClass* value of the port, the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like, for example, *networkTNAAddrVCAT*.
For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
The parameter *localLPIDVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.

For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 0 to 4294967295.
Default: 0.

15. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
The parameter *localLPIDVCAT* is used by ONNS for switching network connections on SONET/SDH layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: Alphanumeric value set.
Range: IPV4, IPV6, or NSAP
Default: NSAP
Restriction: For *modifier* = ES64U, this parameter is not editable.

16. **`networkTNATypeVCAT`** – Network TNA Type on VCAT level
The *networkTNATypeVCAT* identifies the type of the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
The parameter *networkTNATypeVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of eportClass UNI.
This parameter is supported by ports of eportClass EDGE.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: Alphanumeric value set.
Range: IPV4, IPV6, or NSAP
Default: NSAP
Restriction: For *modifier* = ES64U, this parameter is not editable.

17. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given NetworkTNAType. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAAddress", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *networkTNAAddress* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
    Type:
    networkTNAType = IPV4: Four dot separated numeric string (represents 4 Bytes)
    networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    networkTNATyp e= NSAP: Hexadecimal number (represents 20 Bytes)
    Range:
    IPV4: Each numeric string: 0…255
    IPV6: Each hexadecimal number: 0000…FFFF
    NSAP: Hexadecimal number:
    0000000000000000000000000000000000000000 …
    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
    Examples:
    IPV4: 129.12.3.178
    IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF
    Default:
    IPV4: 0.0.0.0
    IPV6: 0-0-0-0-0-0-0-0
    NSAP: 0000000000000000000000000000000000000000
    Restriction: For *modifier* = ES64U, this parameter is not editable.
18. **`networkTNAAddrVCAT`** – Network TNA Address on VCAT level
    The *networkTNAAddrVCAT* identifies the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNATypeVCAT*. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.

The parameter *networkTNAAddrVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type:
networkTNAType = IPV4: Four dot separated numeric string (represents 4 bytes)
networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 bytes)
networkTNAType = NSAP: Hexadecimal number (represents 20 Bytes)
Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000
Restriction: For *modifier* = ES64U, this parameter is not editable.

19. **plcrthr** – Threshold for Partial Loss of Capacity (Receive)
The parameter *plcrthr* specifies the acceptable limit of valid VCG members in receive direction (refer to the *outmbs* parameter of all related VCGTRIB entities) for which the PLCR alarm will not yet be raised. The PLCR alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plcrthr* parameter – 1)] (if *plcrthr* ≥ 2). The PLCR alarm shall never be raised if *plcrthr* is set to 1.
The PLCR alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plcrthr* parameter (or if the number of those VCG members is 0 – in this case the TLCR alarm will be raised instead). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plcrthr* is not supported, then the parameter is omitted.
For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 1, 2, 3, …, 192.
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21

Modifier	Input signal structure	Default value
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192
ES64U	n × 1	n

Restriction: For *modifier* = ES64U, this parameter is not editable.

20. **`plctthr`** – Threshold for Partial Loss of Capacity (Transmit)
    The parameter *plctthr* specifies the acceptable limit of valid VCG members in transmit direction (refer to the *inmbs* parameter of all related VCGTRIB entities) for which the PLCT alarm will not yet be raised. The PLCT alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plctthr* parameter – 1)] (if *plctthr* ≥ 2). The PLCR alarm shall never be raised if *plctthr* is set to 1.
    The PLCT alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plctthr* parameter (or if the number of those VCG members is 0 – in this case the TLCT alarm will be raised instead).
    This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plctthr* is not supported, then the parameter is omitted.
    For *modifier* = ES64U, the parameter is visible with its default value for retrieval, but it cannot be modified.
    Type: numeric value set.
    Range: 1, 2, 3, …, 192.
    Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192
ES64U	n × 1	n

Restriction: For *modifier* = ES64U, this parameter is not editable.

21. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U.

Type: Alphanumeric value set.
Range: DISABLE, STP is disabled, that means, the port is not used to build the spanning tree. ENABLE (default), STP is enabled, that means, the port is used to build the spanning tree.
As VCGs are by default network ports the default value is to enable STP.

22. **`stpcost`** – Spanning Tree Path Cost
    The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
    If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U.
    If the parameter *stpcost* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 to 200000000.
    Default: 20000.
    The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

23. **`stpp`** – Spanning Tree Port Priority
    The parameter *stpp* specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database. This parameter is not supported if the input parameter: *modifier* = ES64U.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *stpp* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.
    Default: 128.

24. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (also known as the implicit reset method)
- TR-RTR (also known as the explicit reset method)

This parameter is not supported if the input parameter: *modifier* = ES64U.
If the parameter *tcamd* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: TRRTR in SDH mode; TR in SONET mode.

25. **`unequiposig`** – Unequipped Tributary Output Signal Rate List
The parameter *unequiposig* specifies the tributary structure and the capacity of the VCG port in output direction. As long as there is no cross-connection applied the specified tributaries are occupied by an unequipped signal. The parameter will implicitly give the maximum capacity of the VCG port, too.
If the input parameter *modifier* is omitted, then the tributary structure depends on interface standard specified for this VCG by *istd* . For the definition of *istd* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).
If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the range of possible tributary structures depends on interface standard specified for this VCG by *istd* and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). The signal will be immediately terminated on the unit. So no unequipped signal is visible to the outside world.
If the input parameter: *modifier* = ES64U, then the parameter *unequiposig* is set automatically according to the value of the *inputsig* parameter.
For *modifier* = ES64U, the parameter *unequiposig* is only retrievable and cannot be changed because it follows always the value of the *inputsig* parameter.
Type: numeric string of the form ***NX***

- ***N*** specifies the count of container type ***X***
- ***X*** specifies the container type (or envelope).

Range:

Values	Modifier	Interface standard	Virtual concatenation mode	Represented signal structure
N = 7, ***X*** = 4	omitted	SDH	not applicable	7 × VC4
N = 7, ***X*** = 4	1GEPL	SDH	ENABLE	7 × VC4
N = 64, ***X*** = 4	10GEPL	SDH	ENABLE	64 × VC4
N = 1, ***X*** = c64	10GEPL	SDH	DISABLE	1 × VC4-64C
N = 21, ***X*** = 1	omitted	SONET	not applicable	21 × STS-1

Values	Modifier	Interface standard	Virtual concatenation mode	Represented signal structure
N = 21, ***X*** = 1	1GEPL	SONET	ENABLE	21 × STS-1
N = 7, ***X*** = 3	1GEPL	SONET	ENABLE	7 × STS-3C
N = 192, ***X*** = 1	10GEPL	SONET	ENABLE	192 × STS-1
N = 64, ***X*** = 3	10GEPL	SONET	ENABLE	64 × STS-3C
N = 1, ***X*** = 192	10GEPL	SONET	DISABLE	1 × STS-192C

For *modifier* = ES64U, the same value set is applicable as for the parameter *inputsig*.
Default:

Modifier	Interface Standard	Default Value
omitted	SDH	74
1GEPL	SDH	74
10GEPL	SDH	644
omitted	SONET	211
1GEPL	SONET	211
10GEPL	SONET	1921
ES64U	SONET	not applicable

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, that means, VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception:
For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.

26. **`vcat`** – Virtual Concatenation
This parameter vcat specifies whether the Ethernet signal is mapped to a number of virtually concatenated tributaries (value = ENABLE) or into a contiguously concatenated tributaries (value = DISABLE). In the latter case, LCAS related parameters become insignificant (values set to default, values cannot be changed). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *vcat* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
- For *modifier* ES64U: ENABLE (default)
- otherwise: DISABLE.
- For *modifier* = ES64U the parameter *vcat* is visible for retrieval, but cannot be modified once the VCG is established.

27. **vmode** – VCG Monitoring Mode
    The parameter *vmode* specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range:
    - MON, for monitored
    - NMON (default), for not monitored.
28. **vpm** – VCG Performance Monitoring
    The parameter *vpm* specifies whether the performance monitoring is enabled or disabled for this port.
    This parameter is not supported if the input parameter: *modifier* = ES64U. If the parameter *vpm* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - DISABLE (default)
    - ENABLE.
29. **vvidls** – Valid VLAN Identifier List (static)
    This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using '&' and '&&' . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (*cnprole* = NETWORK).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *vvidls* is not supported, then the parameter is omitted.
    Type: Alphanumeric string and the character '&'.

Range:

- Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
- In case the list has no members (empty list) –1.

Default: –1.

30. **`wtr`** – Wait To Restore Time
The parameter *wtr* specifies the wait to restore time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it recovers from a faulty state then this tributary is taken in again to transmission by LCAS after the wait-to-restore time. This time is configured on group level, i.e. all tributaries have the same value for the wait-to-restore time. This parameter only affects NE behavior if *lcasmd* = ENABLED.
For *modifier* = ES64U the parameter is visible with default values for retrieval, but cannot be modified.
Type: Alphanumeric value set.
Range: 0, 1, 2, …, 59, 60 [minutes].
Default: 0.
Restriction: For *modifier* = ES64U, this parameter is not editable.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-VCG:1-1-#-#-1-V1::vmode=MON″
;

   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-VCG-10GEPL:1-1-#-#-1-v9::\
    htmr=0,lcasmd=ENABLE,inputsig=1192,plcrthr=192,plctthr=192,\
    unequiposig=1192,vcat=DISABLE,wtr=0″
;

   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-VCG-ES64U:1-1-#-#-22-v9::\
    enetpn=asap66,vmode=MON″
;
```

ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application

The **`ED-VCG-TRANS`** command modifies the properties of a virtual concatenation group for fully transparent applications.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **`ED-VCG-TRANS`***:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values
   - *vall* for Simple_VCGAID
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   is excluded.
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`camd`** – Consequent Action Mode
   This parameter specifies the mode of the consequent action to be executed in egress direction (to the synchronous port) on detected signal fail conditions (dPLM, AI_TSF or detected generic AIS) on the ODU layer
   Type: Alphanumeric value set.
   Range: GENERICAIS (default), MSAIS, LASEROFF
2. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Some VCG related alarms have to be referenced via the ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
3. **`fevpm`** – Far End VCG Performance Monitoring
   The parameter fevpm specifies whether the far end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore far end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
4. **`istd`** – Interface Standard
   This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
   Type: Alphanumeric value set.
   Range:
   - SDH
   - SONET.

   Default: As derived from parameter *istddflt* specified at configuration management, section System RTRV-EQPT.
5. **`nevpm`** – Near End VCG Performance Monitoring
   This parameter specifies whether the near end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore near end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE

6. **`pmvcgfeses`** – Performance Monitoring VCG Far-End SES Threshold
   This parameter describes the ODUkP Far-End SES threshold value. It indicates the number of errored blocks per second to declare a second as severely errored. The default value is defined in G.8201 for OTN. If the interface standard is changed the current default value is not changed.
   Type: numeric value set.
   Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
   Default: ODU1P: 3064
   G.8201 defines block based PM on the ODUkP layer.
7. **`pmvcgneses`** – Performance Monitoring VCG Near-End SES Threshold
   This parameter describes the VCG (actually ODUkP) Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
   Type: numeric value set.
   Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
   Default: ODU1P: 3064
   G.8201 defines block based PM on the ODUkP layer.
8. **`tcamd`** – TCA mode
   This parameter shows the modus of the PM thresholding.
   Two thresholding methods are applicable: TR-only, (also known as the implicit reset method). TR-RTR, (also known as the explicit reset method).
   Type: Alphanumeric value set.
   Range: TR, TRRTR.
   Default: TRRTR in SDH mode; TR in SONET mode.
9. **`transpn`** – Transparent Alarm Severity Assignment Profile Name
   This parameter points to a ASAP profile name of type TRANSPARENCY for the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length
   Default: DEFAULT.
10. **`transtca`** – Transparent TCA Profile Name
    The parameter transtca specifies the name of the threshold crossing alert (TCA) profile used for the VCG..
    Type: Alphanumeric string.
    Range: 1 … 24 characters length
    Default: DEFAULT0
11. **`vmode`** – VCG Monitoring Mode
    This parameter wtr specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range: MON: for monitored; NMON (default): for not monitored.

Output format

If the NE fully complies with the **ED-VCG-TRANS** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed port entity according to the parameters given with this command. The modification is reported by means of an **ED-VCG-TRANS** notification for port (except for parameter *istd*; see below).

Switching of interface standard - Ignore other parameters

If additional parameters beside istd are specified with the **ED-VCG-TRANS** command they will be ignored by the NE. This simple approach has the advantage that all ambiguous situations are avoided and it is not necessary to implement an order or follow a certain sequence when evaluating the parameters.

Switching of interface standard - FM impact

The change of the interface standard of a port from SDH to SONET shall have the following impact related to Fault Management:

- All SDH alarms related to that VCG port is cleared
- The corresponding SONET alarms are raised.

Switching of interface standard - Impact on port parameters

Switching of interface standard

The switching of *istd* should not result in an deletion and recreation of the VCG port. NE internally there are no differences between SDH and SONET.

Switching of interface standard I

In detail the following attributes are influenced by switching of the *istd* from SDH to SONET

- *inputsig* is set to the default value
- *istd* is set to SONET
- *pmvcgfeses* is set to the default value for a SONET port
- *pmvcgneses* is set to the default value for a SONET port

- *transtca* is set to the default value for a SONET port
- *tcamd* is set to the default value for a SONET port
- *unequiposig* is set to its default value.

In detail the following attributes are influenced by switching of the istd from SONET to SDH

- *inputsig* is set to the default value
- *istd* is set to SDH
- *pmvcgfeses* is set to the default value for a SDH port
- *pmvcgneses* is set to the default value for a SDH port
- *transtca* is set to the default value for a SDH port
- *tcamd* is set to the default value for a SDH port
- *unequiposig* is set to its default value.

All other port parameters remain unchanged.

Switching of interface standard - Side effects on tributaries

The switching of *istd* should not result in a deletion and recreation of the tributaries of a VCG port. NE internally there are no differences between SDH and SONET.

In detail the following side-effects apply to the related tributary attributes when switching the *istd* from SDH to SONET:

- *ptca* is set to the default value for a SONET tributary
- *tcamd* is set to the default value for a SONET tributary
- *pmhpneses* is set to the default value for an SONET tributary
- *pmhpfeses* is set to the default value for an SONET tributary
- *sdsfmode* is set to the default value for an SONET tributary
- *hptimais* is set to the default value for an SONET tributary.

In detail the following side-effects apply to the related tributary attributes when switching the istd from SONET to SDH:

- *ptca* is set to the default value for a SDH tributary
- *tcamd* is set to the default value for a SDH tributary
- *pmhpneses* is set to the default value for an SDH tributary
- *pmhpfeses* is set to the default value for an SDH tributary
- *sdsfmode* is set to the default value for an SDH tributary
- *hptimais* is set to the default value for an SDH tributary.

Currently (in this release) not applicable parameters to be ignored.

Switching of interface standard - notification behavior for port attributes

When changing the interface standard istd using the **ED-VCG-TRANS** command all resulting parameter changes are reported by ED-VCG-TRANS notifications. This is valid for the parameter istd itself as well as for all other parameters which might change as a consequence of changing the interface standard.

Switching of interface standard - notification behavior for tributary attributes

When changing the interface standard istd using the **ED-VCG-TRANS** command all resulting tributary parameter changes are reported by ED-VCGTRIB notifications.

Switching of interface standard - notification behavior for active TCA events

If the changing of the interface standard istd to SONET includes a change of tcamd from TR-RTR to TR-only (SONET default value), then all active TCA events of the related entities must be cleared.

Switching of interface standard - PM impact

The change of interface standard for the VCG shall have the following impacts w.r.t. existing PM registers:

- All current bins related to the port is reset.
- All history bins related to the port shall remain unchanged.
- All current bins related to tributaries being a member of the VCG port is reset.
- All history bins related to tributaries being a member of the VCG port shall remain unchanged.

Retrieving the bins after switching the interface standard may result in a strange reporting behavior:

- When switching the interface standard to SONET the periods before the switch contain SDH-related parameters and values even though the port is addressed as a SONET port.
- When switching the interface standard to SDH the periods before the switch contain SONET-related parameters and values even though the port is addressed as a SDH port.

Disabling of PM Monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

- Stop PM counting
- All current bins related to the PM monitor is deleted
- All history bins related to the PM monitor is deleted
- All active TCA events related to the deleted PM monitor have to be cleared if the TCA mode TR-RTR was selected.

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of ’pm enable attributes’.

PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion.

Changing TCA mode from TR-RTR to TR-only

The change of *tcamd* from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Group Notification for VCG tributaries

For efficiency reasons the notifications for VCG tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the used mapping of the associated VCG port, i.e. the Simple_VCGNumAID field in the VCGNumAID is set to

1. 1&&-<N> for sequential range
2. 1&-4 .. &-<N> for non-sequential range

with N is maximal legal VCG tributary number. Example:

The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping) 1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping).

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-VCG-TRANS`** command, too.

Validation rule: Command only allowed for VCG on Transparent Packs

If the command is not applied to a VCG on a transparent pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: crossconnection - interface standard change

Check that the interface standard parameter (istd) is not changed if there is still a cross-connection applied to any tributary of the whole VCG port.

Error code:

`SNVS – Status, Not in Valid State, cross-connections are still existing`

Validation rule: no loopbacks for interface standard change

Check that the interface standard parameter (istd) is not changed when cross-connection loopback is applied to any tributary of the whole VCG port.

Error code: `SNVS – Status, Not in Valid State, a loopback is existing`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

ED-VCG-TRANS notification

Output format

The **ED-VCG-TRANS** notification uses the general format of a REPT DBCHG notification.

Syntax

ED-VCG-TRANS*:aid::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values *vall* for Simple_VCGAID and *usall* for Simple_UnivSlotAID is accepted.

Spec block parameters

1. **camd** – Consequent Action Mode
   This parameter specifies the mode of the consequent action to be executed in egress direction (to the synchronous port) on detected signal fail conditions (dPLM, AI_TSF or detected generic AIS) on the ODU layer
   Type: Alphanumeric value set.
   Range: GENERICAIS (default), MSAIS, LASEROFF
2. **enetpn** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to a tributary ASAP profile name of type ENET for the designated AID.
   Some VCG related alarms have to be referenced via the ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
3. **fevpm** – Far End VCG Performance Monitoring
   The parameter fevpm specifies whether the far end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore far end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
4. **istd** – Interface Standard
   This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
   Type: Alphanumeric value set.

Range:

- SDH
- SONET.

Default: As derived from parameter *istddflt* specified at configuration management, section System RTRV-EQPT.

5. **`nevpm`** – Near End VCG Performance Monitoring
This parameter specifies whether the near end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore near end PM on the ODUkP layer.
Type: Alphanumeric value set.
Range: DISABLE (default), ENABLE

6. **`pmvcgfeses`** – Performance Monitoring VCG Far-End SES Threshold
This parameter describes the ODUkP Far-End SES threshold value. It indicates the number of errored blocks per second to declare a second as severely errored. The default value is defined in G.8201 for OTN. If the interface standard is changed the current default value is not changed.
Type: numeric value set.
Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
Default: ODU1P: 3064
G.8201 defines block based PM on the ODUkP layer.

7. **`pmvcgneses`** – Performance Monitoring VCG Near-End SES Threshold
This parameter describes the VCG (actually ODUkP) Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
Default: ODU1P: 3064
G.8201 defines block based PM on the ODUkP layer.

8. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable: TR-only, (also known as the implicit reset method). TR-RTR, (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: TRRTR in SDH mode; TR in SONET mode.

9. **`transpn`** – Transparent Alarm Severity Assignment Profile Name
   This parameter points to a ASAP profile name of type TRANSPARENCY for the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length
   Default: DEFAULT.
10. **`transtca`** – Transparent TCA Profile Name
    The parameter transtca specifies the name of the threshold crossing alert (TCA) profile used for the VCG..
    Type: Alphanumeric string.
    Range: 1 … 24 characters length
    Default: DEFAULT0
11. **`vmode`** – VCG Monitoring Mode
    This parameter wtr specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range: MON: for monitored; NMON (default): for not monitored.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-VCG-TRANS:1-1-#-#-1-V1::vmode=MON″
;
```

ED-VCGTRIB: Edit virtual concatenation group tributary

The **ED-VCGTRIB** command modifies the properties of a virtual concatenation group tributary.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **ED-VCGTRIB**-*modifier:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
   This parameter specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: STS, VC, STST, VCT.
   To distinguish the fully transparent application from the GE***n***/ES64U application, a letter "T" is appended.
2. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   The access identifier specifies the VCG tributary whose attributes are to be modified.
   Type: AID
   Range: VCGNumAID as defined in AID structure. The values
   - *all* for Simple_VCGNumAID
   - *vall* for Simple_VCGAID
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   is excluded.
4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`brstintvl`**- Bursty Interval
   This parameter specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstintvl* has no affect to NE behavior.
   Type: Integer..
   Range: 2, 3, …, 10 [seconds].
   Default: 7 [seconds].
2. **`brstthr`** – Bursty Error Threshold
   This parameter specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30 [%].
3. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   The parameter *degthr* is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64U circuit packs.
   Type: Integer.
   Range: –5, –6, –7, –8, –9, that means, range is 10^{-9} to 10^{-5}.
   Default: -6
4. **`dexcthr`** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   The parameter dexcthr is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64Ucircuit packs.
   Type: Integer.
   Range: –3, –4, –5 i.e. range is 10^{-5} to 10^{-3}.
   Default: -3.
5. **`exphptrc`** – Expected HP trace identifier
   This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval but cannot be modified.

Type: Hexadecimal number without a preceding “H”.
Range: length, 32 hexadecimal digits; value, specific string acc. ITU/ETSI mode 1.
Default: 89000000000000000000000000000000.
This parameter is not allowed for VCG tributaries located on ES64U packs.
This specification is necessary due to hardware restrictions of the Gauss device.

6. **feppm** – Far End Path PM Enable
   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
   The parameter *feppm* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
   For VCG tributaries located on ES64U packs, this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE.
   Default: DISABLE.
   This parameter is not allowed for VCG tributaries located on ES64U packs.

7. **hptimais** – HP Trace Identifier Mismatch AIS Insertion
   This parameter specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: DISABLE , ENABLE.
   Default: The default value depends on interface standard specified by istd for the VCG owning this tributary. For definition of *istd* refer to “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078) Output parameters. DISABLE, if *istd* = SONET; ENABLE, if *istd* = SDH.

8. **hptimmon** – HP Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE.

9. **inmbs** – Input Membership
   This parameter specifies if the tributary shall take part to the transmission of its owning VCG in input direction. Input direction in this case means from the LCAS source on the GE1 pack to the cross-connect pack XC***n***. The addressing scheme of the tributaries by using VCGNumAIDs is ensuring that only legal AIDs are used. So no consideration about the “illegal” tributaries (for example, 1-1-#-#-1-1-3 for VC-4) is necessary. The parameter *inmbs* is only valid for modifiers different from STST and VCT, that means, is omitted in the command response for STST and

VCT.
The system shall not evaluate the parameter *inmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
Type: Alphanumeric value set.
Range: DISABLE , ENABLE.
Default:

- for modifiers STST or VCT: ENABLE
- for all other modifiers: DISABLE.

The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.

10. **`neppm`** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
The parameter *neppm* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
Type: Alphanumeric value set.
Range:
    - DISABLE (default)
    - ENABLE.

    This parameter is not allowed for VCG tributaries located on ES64U packs.

11. **`outmbs`** – Output Membership
This parameter specifies if the tributary shall take part to the transmission of its owning VCG in output direction. Output direction in this case means from the cross-connect pack XCn to the LCAS sink on the GE***n*** pack. The parameter *outmbs* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
The system shall not evaluate the parameter *outmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
Type: Alphanumeric value set.
Range:
    - DISABLE
    - ENABLE.

    The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.

12. **`pmhpfeses`** – PM HP Far End SES Threshold
    This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078). If *istd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.
    The parameter *pmhpfeses* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
    Type: Integer.
    Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
    Default: 2400.
13. **`pmhpneses`** – PM HP Near End SES Threshold
    This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078). If *istd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
    The parameter *pmhpneses* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
    Type: Integer.
    Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
    Default: 2400.
14. **`ptca`** – Path TCA Profile Pointer
    This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.
    The parameter *ptca* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
    Type: Alphanumeric string.
    Range: 1 … 24 characters.
    Default: DEFAULT0.

15. **`ptpn`** – Path Terminating ASA Profile Name
    This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: 1 … 24 characters.
    Default: DEFAULT.
16. **`ptrcrfmt`** – Path Trace Read Format
    This parameter specifies the format of the path trace (J1 byte) to be received (sink).
    The parameter is not allowed for VCG tributaries located on GE10PL1 packs.
    The parameter is not allowed for VCG tributaries located on ES64U packs. This specification is necessary due to hardware restrictions of the Gauss device.
    Type: numeric string.
    Range:
    - 2, for non-specific repeating byte
    - 16, for 16 byte specific string
    - 64, for 64 byte string.

    Default: 2.
17. **`ptrcwfmt`** – Path Trace Write Format
    This parameter specifies the format of the path trace (J1 byte) to be sent (source).
    The parameter is not allowed for VCG tributaries located on GE10PL1 packs.
    The parameter is not allowed for VCG tributaries located on ES64U packs. This specification is necessary due to hardware restrictions of the Gauss device.
    Type: numeric string.
    Range:
    - 2, for non-specific repeating byte
    - 16, for 16 byte specific string
    - 64, for 64 byte string.

    Default: 2.
18. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    This parameter specifies the determination mode of degraded defect for transmission.
    The parameter *sdsfmode* is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64U circuit packs.
    Type: Alphanumeric value set.
    Range: BURST or POISSON.
    Default: For VCG tributaries located on a GE10PL1 pack, the default is: BURST. For VCG tributaries located on other packs, the default value depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to **`RTRV-VCG`** Output Parameters. POISSON, if *istd* = SONET; BURST, if *istd* = SDH

19. **tcamd** – TCA mode
    This parameter is only available for SDH mode. This means the modifier must be VC. The parameter shows the modus of the PM thresholding. Two thresholding methods are applicable for SDH: TR-only (also known as the implicit reset method), TR-RTR (also known as the explicit reset method). For SONET the parameter *tcamd* does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
    The parameter *tcamd* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
    Type: Alphanumeric value set.
    Range: TR, TRRTR.
    Default: SDH mode: TRRTR. SONET mode: TR (internal). For SONET the parameter tcamd does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
20. **tmonmd** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.
    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect.
    Type: Alphanumeric value set.
    Range: MON or NMON (default).
21. **trc** – Outgoing Path Trace
    This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
    Type: Hexadecimal number without a preceding “H”.
    Range: the range depends on the parameter ptrcwfmt specified in **RTRV-VCGTRIB** Output parameters length
    - 32, hexadecimal digits, if *ptrcwfmt* = 16
    - 128, hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt* = 16; 64 byte string, if *ptrcwfmt* = 16.

Default: Depending on the default value of *ptrcwfmt*:

- 00hex, if *ptrcwfmt* = 2.
- 89hex followed by 15 bytes 00hex, if *ptrcwfmt* = 16.
- 64 bytes 00hex, if *ptrcwfmt* = 64.

The parameter is not allowed for VCG tributaries located on ES64U packs.
This specification is necessary due to hardware restrictions of the Gauss device.

22. **tslmmon** – Trail Signal Label Mismatch Monitor
This parameter specifies the mode of the TSL mismatch detection.
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: Alphanumeric value set.
Range:
    - DISABLE
    - ENABLE (default).

The parameter is not allowed for VCG tributaries located on ES64U packs.
This specification is necessary due to hardware restrictions of the Gauss device.

Output format

If the NE fully complies with the **ED-VCGTRIB** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **ED-VCGTRIB** generates a **REPT DBCHG** message.

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed tributary entity according to the parameters given with this command.

The parameters *brstintvl*, *brstthr*, *degthr* and *dexcthr* is modifiable also if they are not meaningful at the moment. The modification is reported by means of an ED-VCGTRIB notification.

Dependencies

brstintvl and *brstthr* are only meaningful if sdfsmode is set to BURST. *degthr* and *dexcthr* are only meaningful if sdfsmode is set to POISSON.

Update affected entities

A change of the value of one of the parameters *inmbs* and *outmbs* shall trigger the report of *inmbst* and *outmbst* of the corresponding VCGTRIB via an autonomous notification.

Update affected entities - *hptimais* for VCG tributaries of a transparent VCG

If the value of hptimais is changed for one VCG tributary of a transparent VCG (modifiers STST and VCT), then the parameter hptimais are updated to the same value for all VCG tributaries of the same transparent VCG. The update of *hptimais* is notified via ED-VCGTRIB notifications for each VCG tributary.

Supply outgoing J1 trace value and format together

When changing the format of the outgoing J1 trace *ptrcwfmt*, the parameter trc specifying the outgoing J1 trace shall always be supplied within the same TL1-command to always have a consistent set of trace format and value. When the format of the outgoing J1 trace *ptrcwfmt* is set to 2 (non-specific repeating byte) then the outgoing J1 trace (mandatory parameter trc) specified in the command is ignored.

Change of J1 expected format - FM impact

When the expected J1 format (*ptrcrfmt*) is changed to 16 (16 byte specific string) while the TI Mismatch detection is enabled, then the TI Mismatch alarm is updated according to the provisioned ecpected J1 (exphptrc) and the currently received J1 (ptrc). When the expected J1 format is changed away from 16, while the TI Mismatch alarm is active, then the TI Mismatch alarm is cleared.

Not applicable parameters for the modifiers STST and VCT

For the command modifiers STST and VCT the following parameters is not alterable:

- **`brstintvl`**
- **`brstthr`**
- **`degthr`**
- **`dexcthr`**
- **`feppm`**
- **`inmbs`**
- **`inmbst`**
- **`neppm`**
- **`outmbs`**
- **`outmbst`**
- **`pmhpfeses`**
- **`pmhpneses`**
- **`ptca`**

- **sdsfmode**
- **tcamd**

See also the invalid declaration of these parameters for **RTRV-VCGTRIB** for these modifiers.

Disabling of PM Monitor

The disabling of a PM monitor shall have the following impacts w.r.t. existing PM registers:

- Stop PM counting
- All current bins related to the PM monitor is deleted
- All history bins related to the PM monitor is deleted

The term “PM monitor” has the same semantics as the term “PM package”. The number of PM monitors is determined by means of the number of ’pm enable attributes’.

PM monitors can be disabled explicitly by means of the ’pm enable’ attribute or are implicitly disabled as part of a port/tributary deletion.

Additional to the above listed impacts the disabling of a PM monitor shall have the following impacts: the system has to clear all active TCA events of the deleted PM monitor if TCA mode TR-RTR was selected .

Changing TCA mode from TR-RTR to TR-only

The change of tcamd from TR-RTR (also known as the explicit reset method) to TR-only (also known as the implicit reset method) has the impact, that all active TCA events of the related entity must be cleared.

Threshold settings for all VCG tributaries related to one VCG

Setting one of the parameters *brstintvl*, *brstthr*, *degthr*, *dexcthr*, *sdsfmode* for one VCG Tributary causes that this parameter is set to the same value for all other VCG tributaries related to the same VCG. The respective notifications are sent.

Refer to the parameter definitions in “RTRV-VCGTRIB: Retrieve virtual concatenation group tributary ” (p. 3-3109) for the restrictions using the modifiers STST and VCT.

64 byte path trace setting on GE10PL1 pack for all VCG tributaries related to one VCG

Setting the outgoing path trace (ptrc) for the first VCG tributary of a VCG on a GE10PL1 pack, while the outgoing path trace format (ptrcwfmt) is set to 64 byte format, shall set the outgoing path trace of all VCG tributaries of the VCG to the same value as the first VCG tributary.

Example input/output

```
ED-VCGTRIB-VC:LT-UNITE:1-1-#-#-1-v1-1:123456:::tmonmd=NMON;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-VCGTRIB`** command.

Validation rule: Check *modifier* against circuit pack type

If the command is applied

- with *modifier* = VC or STS to a VCG Tributary on a Transparent pack or
- with *modifier* = VCT or STST to a VCG Tributary on a GE***n*** or ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: ASAP profile exists

Check of the existence of the ASAP profile.

Error code: `IDNV – Input, Data Not Valid, ASAP profile does not exist`

Validation rule: TCA profile exists

Check of the existence of the TCA profile.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Validation rule: supply of transmitted path trace

Check that a transmit path trace (trc) is supplied, too, if the path trace mode is changed.

Error code:

`IDNC – Input, Data, Not Consistent, a new trc must also be supplied for a path trace mode change`

Validation rule: transmitted path trace length

Check that a length of the transmit path trace (*trc*) fits to the length given by path trace mode, i.e. write format (*ptrcwfmt*).

Error code:

`IDNC – Input, Data, Not Consistent, trc length must match the given path trace mode`

Validation rule: Path trace monitoring on GE10PL1 only for first VCG tributary of a VCG

If the VCG tributary is located on a GE10PL1 pack, then the system shall only allow to set the parameters hptimmon and hptimais to ENABLE only for the first VCG tributary of a VCG. I.e. only for VCG tributary AID number = 1.

Error code:
`SDNC – Status, Data Not Consistent, Not allowed on GE10PL1 pack`

Validation rule: Path trace not supported on ES64U

If the VCG tributary is located on an ES64U pack, then the system does not allow to set the parameters *exphptrc*, *ptrcrfmt*, *ptrcwfmt*, and *trc*.

This validation rule is necessary due to hardware restrictions.

Error code: `SDNC – Status, Data Not Consistent, Not allowed on ES64U pack`

Validation rule: Trail signal label mismatch monitor not supported on ES64U

If the VCG tributary is located on an ES64U pack, then the system does not allow to set the parameter *tslmmon*

This validation rule is necessary due to hardware restrictions.

Error code: `SDNC – Status, Data Not Consistent, Not allowed on ES64U pack`

Validation rule:

If the VCG tributary is located on an ES64U pack, then the system does not allow to set the parameters *feppm* and *neppm*.

Error code: `SDNC – Status, Data Not Consistent, Not allowed on ES64U pack`

Validation rule: 64 byte path trace setting on GE10PL1 only for first VCG tributary of a VCG

If the VCG tributary is located on a GE10PL1 pack and the parameter *ptrcwfmt* has the value 64, then the system shall only allow to set the parameter *trc* only for the first VCG tributary of a VCG. I.e. only for VCG tributary AID number = 1.

Error
code:`SDNC – Status, Data Not Consistent, Not allowed on GE10PL1 pack`

Validation rule: Maximum number of enabled PM monitors

Check that the number of enabled PM monitors is less than the maximum number of supported enabled PM monitors (2000) before a new PM monitor is enabled.

Error code:
`SSRE – Status, System Resources Exceeded, limit of PM monitors exceeded`.

Since several monitor points can be enabled/disabled within one command, the **`ED`** command may only fail if the validation rule is not met after all PM enable attributes would have been updated as specified in the corresponding ED request.

Example: The **`ED`** command requests that 1 monitor point is enabled and 1 monitor points is disabled and currently 2000 packages are enabled. This command should complete.

Validation rule: Correct state of tributary for being added (Sink)

Check that the tributary to be added for transmission in sink direction is in the inactive state (NA, not applicable).

Error code: `ICNV – Input, Command Not Valid, outmbst not NA`

For the detailed description of the LCAS protocol refer to SRD_Data_VirtualConcatenation. Only applicable for modifiers different from STST and VCT.

Validation rule: Correct state of tributary for being added (Source)

Check that the tributary to be added for transmission in source direction is in the inactive state (NA, not applicable).

Error code: `ICNV – Input, Command Not Valid, inmbst not NA`

For the detailed description of the LCAS protocol refer to SRD_Data_VirtualConcatenation. Only applicable for modifiers different from STST and VCT.

Validation rule: LCAS failure of protocol present in sink direction

Check if a tributary is going to be added for transmission under active LCAS that no failure of the LCAS protocol is present in receive direction.

Error code:

`ICNV – Input, Command Not Valid, Protocol Failure in Receive Direction`

For the detailed description of the LCAS protocol refer to SRD_Data_VirtualConcatenation. Only applicable for modifiers different from STST and VCT.

Validation rule: LCAS failure of protocol present in source direction

Check if a tributary is going to be added for transmission under active LCAS that no failure of the LCAS protocol is present in transmit direction.

Error code:

`ICNV – Input, Command Not Valid, Protocol Failure in Transmit Direction`.

For the detailed description of the LCAS protocol refer to SRD_Data_VirtualConcatenation.

Only applicable for modifiers different from STST and VCT.

Validation rule: Enabling of inmbs and outmbs not allowed while vcat disabled

The system shall not allow to set the Input Membership or Output Membership (inmbs or outmbs) to ENABLE for a VCG Tributary while the Virtual Concatenation (vcat) of the related VCG is set to DISABLE.

Error code:

`SNVS – Status, Not in Valid State, Not allowed if VCAT disabled`

Check CCB against istd of related VCG

See definition of Error responses in "RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109).

Related TL1 messages

- "RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109)
- "OPR-VCGTRIB: Operate virtual concatenation group tributary" (p. 3-2082).

ED-VCGTRIB notification

Output format

The **ED-VCGTRIB** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

The **ED-VCGTRIB** notification has the following specific syntax:

ED-VCGTRIB:*aid,aidtype::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG tributary whose attributes are reported.
   Type: AID.
   Range: VCGNumAID as defined in AID structure with exception of value all.
2. **aidtype** – Access Identifier Type
   This parameter specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: STS, VC, STST, VCT.
   To distinguish the fully transparent application from a GE application, a letter ″T″ is appended. There is no distinction between VCGTRIB entities of the GE1 pack from the GE10PL1 pack needed. In particular, the transmission rate in case the parameter *modifier* = STS needs to be derived from the parameters inputsig and unequiposig of the related VCG entity.

Spec block parameters

1. **brstintvl**- Bursty Interval
   This parameter specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstintvl* has no affect to NE behavior.
   Type: Integer..
   Range: 2, 3, …, 10 [seconds].
   Default: 7 [seconds].
2. **brstthr** – Bursty Error Threshold
   This parameter specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30 [%].

3. **`crslpbkstat`** – Cross-connect Loopback Status
   This parameter reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to “OPR-LPBK: Operate cross-connection loopback” (p. 3-2035).
   Type: Alphanumeric value set.
   Range: YES, for enabled loopback; NO, for disabled loopback.
   Default: NO.
4. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   The parameter *degthr* is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64U circuit packs.
   Type: Integer.
   Range: –5, –6, –7, –8, –9, that means, range is 10^{-9} to 10^{-5}.
   Default: -6
5. **`dexcthr`** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   The parameter dexcthr is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64Ucircuit packs.
   Type: Integer.
   Range: –3, –4, –5 i.e. range is 10^{-5} to 10^{-3}.
   Default: -3.
6. **`exphptrc`** – Expected HP trace identifier
   This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval but cannot be modified.
   Type: Hexadecimal number without a preceding “H”.
   Range: length, 32 hexadecimal digits; value, specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000.
   This parameter is not allowed for VCG tributaries located on ES64U packs.
   This specification is necessary due to hardware restrictions of the Gauss device.

7. **feppm** – Far End Path PM Enable
   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
   The parameter *feppm* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
   For VCG tributaries located on ES64U packs, this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE.
   Default: DISABLE.
   This parameter is not allowed for VCG tributaries located on ES64U packs.
8. **hptimais** – HP Trace Identifier Mismatch AIS Insertion
   This parameter specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: DISABLE , ENABLE.
   Default: The default value depends on interface standard specified by istd for the VCG owning this tributary. For definition of *istd* refer to "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078) Output parameters. DISABLE, if *istd* = SONET; ENABLE, if *istd* = SDH.
9. **hptimmon** – HP Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE.
10. **inmbs** – Input Membership
    This parameter specifies if the tributary shall take part to the transmission of its owning VCG in input direction. Input direction in this case means from the LCAS source on the GE1 pack to the cross-connect pack XC***n***. The addressing scheme of the tributaries by using VCGNumAIDs is ensuring that only legal AIDs are used. So no consideration about the "illegal" tributaries (for example, 1-1-#-#-1-1-3 for VC-4) is necessary. The parameter *inmbs* is only valid for modifiers different from STST and VCT, that means, is omitted in the command response for STST and VCT.
    The system shall not evaluate the parameter *inmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
    Type: Alphanumeric value set.
    Range: DISABLE , ENABLE.

Default:

- for modifiers STST or VCT: ENABLE
- for all other modifiers: DISABLE.

The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.

11. **inmbst** – Input Member Status

This parameter reports the status of the tributary related to its VCG membership in input direction. The parameter *inmbst* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
Type: Alphanumeric value set.

Range:

- for modifiers STST or VCT always ACTIVE is reported
- for all other modifiers:
- NA, for not applicable, i.e. if the trib. is no member of the VCG or if virtual concatenation is disabled
- ADD, the tributary is in the LCAS state ″Wait-for-Add″, ″Add″ or ″Wait-for-ACK″
- ACTIVE, the tributary is member of the VCG and working
- FAILED, the tributary is member of the VCG but taken off by LCAS (incoming failed condition for the sink tributary)
- UNAVAILABLE, for not available, i.e. the trib. is member of the VCG but the current data is not retrievable for any reason.

  The state FAILED can only be reported if *lcasmd* is set to ENABLE for the VCG owning this tributary. For parameter *lcasmd* refer to the Output parameters section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).

  For VCG tributaries located on ES64U packs only ″NA″ will be reported.

12. **neppm** – Near End Path PM Enable

This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
The parameter *neppm* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
Type: Alphanumeric value set.

Range:

- DISABLE (default)
- ENABLE.

This parameter is not allowed for VCG tributaries located on ES64U packs.

13. **`outmbs`** – Output Membership
    This parameter specifies if the tributary shall take part to the transmission of its owning VCG in output direction. Output direction in this case means from the cross-connect pack XCn to the LCAS sink on the GE***n*** pack. The parameter *outmbs* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    The system shall not evaluate the parameter *outmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
    Type: Alphanumeric value set.
    Range:
    - DISABLE
    - ENABLE.

    The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.
14. **`outmbst`** – Output Member Status
    This parameter reports the status of the tributary related to its VCG membership in output direction. The parameter *outmbst* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    Type: Alphanumeric value set.
    Range:
    - for modifiers STST or VCT always ACTIVE is reported, for all other modifiers:
    - NA, for not applicable, i.e. if the trib. is no member of the VCG or if virtual concatenation is disabled
    - ADD, the tributary is in the LCAS state ″Wait-for-Add″ or ″Wait-for-Data″
    - ACTIVE, the tributary is member of the VCG and working
    - FAILED, the tributary is member of the VCG but not working due to a failure condition. Includes period during ″Hold-Off″.
    - WTR, failure condition has cleared, tributary is in wait to restore condition.
    - UNAVAILABLE, for not available, i.e. the trib. is member status is not retrievable for any reason.

      The state WTR can only be reported if lcasmd is set to ENABLE for the VCG owning this tributary. For parameter lcasmd refer to RTRV-VCG Output Parameters. For a detailed explanation of the ADD state in the LCAS protection scheme, please refer to SRD_DATA_Virtual_Concatenation.
      For VCG tributaries located on ES64U packs only ″NA″ will be reported.
15. **`pmhpfeses`** – PM HP Far End SES Threshold
    This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *istd* for the VCG owning this

tributary. For definition of *istd* refer to the output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078). If *istd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.
The parameter *pmhpfeses* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
Default: 2400.

16. **pmhpneses** – PM HP Near End SES Threshold
This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078). If *istd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
The parameter *pmhpneses* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
Default: 2400.

17. **ptca** – Path TCA Profile Pointer
This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.
The parameter *ptca* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Alphanumeric string.
Range: 1 … 24 characters.
Default: DEFAULT0.

18. **ptpn** – Path Terminating ASA Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: 1 … 24 characters.
Default: DEFAULT.

19. **`ptrcrfmt`** – Path Trace Read Format
    This parameter specifies the format of the path trace (J1 byte) to be received (sink).
    The parameter is not allowed for VCG tributaries located on GE10PL1 packs.
    The parameter is not allowed for VCG tributaries located on ES64U packs. This specification is necessary due to hardware restrictions of the Gauss device.
    Type: numeric string.

    Range:

    - 2, for non-specific repeating byte
    - 16, for 16 byte specific string
    - 64, for 64 byte string.

    Default: 2.

20. **`ptrcwfmt`** – Path Trace Write Format
    This parameter specifies the format of the path trace (J1 byte) to be sent (source).
    The parameter is not allowed for VCG tributaries located on GE10PL1 packs.
    The parameter is not allowed for VCG tributaries located on ES64U packs. This specification is necessary due to hardware restrictions of the Gauss device.
    Type: numeric string.

    Range:

    - 2, for non-specific repeating byte
    - 16, for 16 byte specific string
    - 64, for 64 byte string.

    Default: 2.

21. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    This parameter specifies the determination mode of degraded defect for transmission.
    The parameter *sdsfmode* is not allowed for modifiers STST and VCT and for VCG tributaries located on GE10PL1 or ES64U circuit packs.
    Type: Alphanumeric value set.
    Range: BURST or POISSON.
    Default: For VCG tributaries located on a GE10PL1 pack, the default is: BURST.
    For VCG tributaries located on other packs, the default value depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to **`RTRV-VCG`** Output Parameters. POISSON, if *istd* = SONET; BURST, if *istd* = SDH

22. **`tcamd`** – TCA mode
    This parameter is only available for SDH mode. This means the modifier must be VC. The parameter shows the modus of the PM thresholding. Two thresholding methods are applicable for SDH: TR-only (also known as the implicit reset method), TR-RTR (also known as the explicit reset method). For SONET the parameter *tcamd* does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
    The parameter *tcamd* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.

For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: SDH mode: TRRTR. SONET mode: TR (internal). For SONET the parameter tcamd does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

23. **`tmonmd`** – Tributary Monitoring Mode
This parameter specifies the monitoring mode for the tributary.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect.
Type: Alphanumeric value set.
Range: MON or NMON (default).

24. **`trc`** – Outgoing Path Trace
This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: Hexadecimal number without a preceding “H”.
Range: the range depends on the parameter ptrcwfmt specified in **`RTRV-VCGTRIB`** Output parameters length

- 32, hexadecimal digits, if *ptrcwfmt* = 16
- 128, hexadecimal digits, if *ptrcwfmt* = 64
- value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt* = 16; 64 byte string, if *ptrcwfmt* = 16.

Default: Depending on the default value of *ptrcwfmt*:

- 00hex, if *ptrcwfmt* = 2.
- 89hex followed by 15 bytes 00hex, if *ptrcwfmt* = 16.
- 64 bytes 00hex, if *ptrcwfmt* = 64.

The parameter is not allowed for VCG tributaries located on ES64U packs.
This specification is necessary due to hardware restrictions of the Gauss device.

25. **`tslmmon`** – Trail Signal Label Mismatch Monitor
This parameter specifies the mode of the TSL mismatch detection.
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: Alphanumeric value set.

Range:

- DISABLE
- ENABLE (default).

The parameter is not allowed for VCG tributaries located on ES64U packs. This specification is necessary due to hardware restrictions of the Gauss device.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ED-VCGTRIB:1-1-#-#-1-V1-1,STS::outmbs=ENABLE″
;
```

ED-VRTSW: Edit virtual switch

The **ED-VRTSW** command modifies the properties of a virtual switch.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No

Input format

Command syntax: **Ed-VRTSW**:*tid*:*aid*:*ctag*::*vrtswid*:*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
2. **aid** – Access Identifier
   The access identifier specifies the slot containing a GE1 pack.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure.
   The values
   - all and
   - usall for Simple_UnivSlotAID
   is accepted.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
4. **vrtswid** – Virtual Switch Identifier
   This parameter identifiers the instance of the virtual switch.
   - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
   - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.
   Type: Integer.
   Range: 1 – Currently this is the only allowed value.
   Default: 1.

Spec block parameters

1. **`ethertype`** – VLAN tag protocol ID
   This parameter specifies the VLAN tag protocol ID necessary for interworking purposes. This parameter is used to control whether the virtual switch operates in IEEE802.1Q tagging mode (i.e. set to 0x8100) or in transparent tagging mode (i.e. set to value different to 0x8100).
   Type: Hexadecimal value.
   Range: 0x0600 .. 0xFFFF.

   The following values have a specific meaning:

   - 0x8100: virtual switch operates in IEEE802.1Q tagging mode
   - any other value: virtual switch operates in transparent tagging mode.

   Default: 0x8100.
2. **`gvrpmode`** – GVRP mode
   The parameter *gvrpmode* allows to control the dynamic VLAN registration mode.
   Type: Alphanumeric value set.
   Range: ENABLED, GVRP is enabled. DISABLED (default), GVRP is disabled.
3. **`stbrpriority`** – Spanning tree bridge priority
   This parameter specifies the priority of the bridge to be used by the Spanning Tree Protocol.
   Type: Integer.
   Range: 0 to 65535.
   The stepping as recommended by the standard (steps of 4096 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 61440″ is not done by the NE.
   Default: 32768.
4. **`stfdtm`** – Spanning Tree forward delay timer
   The parameter *stfdtm* specifies the rSTP-related bridge forward delay timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 255.
   Default: 15.

5. **`sthellotm`** – Spanning Tree hello timer
   The parameter *sthellotm* specifies the rSTP-related bridge hello timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 225.
   Default: 2.
6. **`stmaxage`** – Spanning Tree maximum age timer
   The parameter *stmaxage* specifies the rSTP-related maximum PDU transmission delay timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 255.
   Default: 20.

Output format

If the NE fully complies with the ED-VRTSW request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of.

Detailed behavior description

The **`ED-VRTSW`** command generates a REPT DBCHG message.

The command modifies the attributes (listed in the spec_block parameters part of this command) of the addressed virtual switch entity according to the parameters given with this command.

The modification is reported by means of an ED-VRTSW notification.

Effect of *ethertype* on virtual switch operation mode

Setting the parameter *ethertype* to a value different from 0x8100 implicitly changes the mode of operation of the virtual switch to transparent tagging.

Reset to default values when changing *ethertype*

When changing ethertype to a value different from 0x8100 the parameter vvidls of all EPORTs and VCGs with cnprole=CUSTOMER is set to its default value.

Example input/output

```
ED-VRTSW:LT-UNITE:1-1-#-#-
      1:123457::1:ethertype=0x8100,gvrpmode=DISABLED,stbrpriority=28672,stfdtm=20,sthellotm=
```

```
   LT-UNITE 00-11-11 11:11:00
M  123457 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ED-EPORT** command.

Related TL1 messages

- “RTRV-VRTSW: Retrieve virtual switch” (p. 3-3125)

ED-VRTSW notification

Output format

The **ED-VRTSW** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Command syntax: **ED-VRTSW***:aid:vrtswid:spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the slot containing a GE1 pack to be operated on.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure with the exception of the value all and usall.
2. **vrtswid** – Virtual Switch Identifier
   This parameter identifiers the instance of the virtual switch.
   - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
   - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

   Type: Integer.
   Range: 1 – Currently this is the only allowed value.
   Default: 1.

Spec block parameters

1. **ethertype** – VLAN tag protocol ID
   This parameter specifies the VLAN tag protocol ID necessary for interworking purposes. This parameter is used to control whether the virtual switch operates in IEEE802.1Q tagging mode (i.e. set to 0x8100) or in transparent tagging mode (i.e. set to value different to 0x8100).
   Type: Hexadecimal value.
   Range: 0x0600 .. 0xFFFF.

The following values have a specific meaning:

- 0x8100: virtual switch operates in IEEE802.1Q tagging mode
- any other value: virtual switch operates in transparent tagging mode.

Default: 0x8100.

2. **`gvrpmode`** – GVRP mode
   The parameter *gvrpmode* allows to control the dynamic VLAN registration mode.
   Type: Alphanumeric value set.
   Range: ENABLED, GVRP is enabled. DISABLED (default), GVRP is disabled.
3. **`rstpmd`** – Rapid Spanning Tree Protocol Mode
   This parameter allows to enable or disable the Rapid Spanning Tree Protocol for the given virtual switch.
   Type: Alphanumeric value set.
   Range: DISABLE (rSTP is disabled, i.e. the virtual switch only supports normal STP (IEEE802.1D)); ENABLE (rSTP is enabled, i.e. the virtual switch also supports rSTP (IEEE802.1w)).
   Default: DISABLE
4. **`stbrpriority`** – Spanning tree bridge priority
   This parameter specifies the priority of the bridge to be used by the Spanning Tree Protocol.
   Type: Integer.
   Range: 0 to 65535.
   The stepping as recommended by the standard (steps of 4096 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 61440″ is not done by the NE.
   Default: 32768.
5. **`stfdtm`** – Spanning Tree forward delay timer
   The parameter *stfdtm* specifies the rSTP-related bridge forward delay timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 255.
   Default: 15.

6. **sthellotm** – Spanning Tree hello timer
   The parameter *sthellotm* specifies the rSTP-related bridge hello timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 225.
   Default: 2.
7. **stmaxage** – Spanning Tree maximum age timer
   The parameter *stmaxage* specifies the rSTP-related maximum PDU transmission delay timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 255.
   Default: 20.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:ED-VRTSW:1-1-#-#-1:1:gvrpmode=DISABLED”
;
```

ENT-ASAP-PROF: Enter alarm severity assignment profile

The **`ENT-ASAP-PROF`** command is used to create a new ASAP and to define the name of the ASAP.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **`ENT-ASAP-PROF`**:*tid*::*ctag*::*pftype*,*pfname*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`ctag`** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **`pftype`** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring

- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

4. **pfname** – ASAP Profile Name
This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
Range: length 1-24 characters.
For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Output format

The following example shows the successful completion of the **ENT-ASAP-PROF** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

This command creates a new ASAP of ASAP type *pftype* and profile name *pfname*. All ASAP parameters defined for the specific *pftype* will have default values; that means, the values are derived from the actual values of the default ASAP (with name DEFAULT).

The maximum total number of ASAPs in an NE includes the predefined “DEFAULT” ASAPs. (Note: The maximum number of ASAPs is defined per system not per *pftype*.)

The modification is reported via the **ENT-ASAP-PROF** notification.

Example input/output

The following example shows the successful completion of the **ENT-ASAP-PROF** command by the NE:

```
ENT-ASAP-PROF:LT-UNITE-123::123456::ENET,att;

   LT-UNITE 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ENT-ASAP-PROF`** command.

Validation rule: Duplicate *pfname*

The *pfname* is already used for the specified *pftype*.

Error code: `IEAE – Input, Entity Already Exists, duplicate profile name`

Validation rule: Maximum number of profiles

The maximum number of profiles is exceeded.

Error code:
`SSRE – Status, System Resources Exceeded, limit of profiles exceeded`

Related TL1 messages

- “DLT-ASAP-PROF: Delete alarm severity assignment profile” (p. 3-134)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “RTRV-ASAP-PROF: Retrieve ASAP profile” (p. 3-2249)
- “RTRV-ASAP-ASGNMT: Retrieve ASAP assignment” (p. 3-2244).

ENT-ASAP-PROF notification

Output format

Syntax: The **ENT-ASAP-PROF** notification has the following specific syntax:

ENT-ASAP-PROF::*pftype*,*pfname*:*spec_block*

Output parameters

Command parameters

1. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring

- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

2. **`pfname`** – ASAP Profile Name
This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
Range: length 1-24 characters.
For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Spec block parameters

1. Alarm Severity Profile parameters
The spec block contains ASAP parameters. Each ASAP parameter defines for a probable cause (*condtype*) the assigned severity which will be reported when the alarm appears. If for one specific probable cause a service affecting (sa) or non service affecting (nsa) alarm can be reported then for each the sa-alarm and nsa-alarm an ASAP parameter exists.
The severity indicates the urgency of an alarm. The severity of a sa-alarm must be more urgent that of a nsa-alarm for the same *condtype*.
The description of the ASAP parameters, the relation to the profile types and the default values are described in: “Alarms, events, ASAPs” (p. 6-4).
All parameters have the same type and value range.
Type: Alphanumeric value set.
Range:

- CR – Critical (ITU / Telcordia)
- MJ – Major (ITU / Telcordia)
- MN – Minor (ITU / Telcordia)
- NA – Warning (ITU) / Not-Alarmed (Telcordia)
- NR – Non-Alarmed (ITU) / Not-Reported (Telcordia).

Example: Concrete ASAP parameters are *sa_eqpt* and *nsa_eqpt*. Both refers to the probable cause (*condtype*) EQPT and ASAP type EQUIPPROT. *sa_eqpt* contains the assigned severity for the service affecting alarm of probable cause EQPT and *nsa_eqpt* for the non service affecting alarm. *sa_eqpt* can have the concrete value CR and *nsa_eqpt* MN.

Example output

The example shows a normal notification after an ENT-ASAP-PROF command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
```

```
”TIME=08-00-12,DATE=00-01-01:ENT-ASAP-PROF::ENET,att:sa_vcg_loa=MJ,nsa_vcg_
    loa=MN,sa_vcg_ssf=MJ,nsa_vcg_ssf=MN,sa_vcg_plm=MN,nsa_vcg_plm=NA,sa_gfp_
    lof=CR,nsa_gfp_lof=MJ,sa_lan_los=CR,nsa_lan_los=MJ,sa_lan_anm=MN,nsa_lan_
    anm=NA”
;
```

ENT-BANNER

The **ENT-BANNER** command can be initiated by a user to provision a proprietary notice to enhance system security. The intent is to warn would be intruders against illegal intrusion into the system.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No

Input format

Command syntax: **ENT-BANNER** :*tid*::*ctag*:::*spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Target Identifier. Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **line1** – Line 1 of the Customer Provisionable Banner.
   Setting line1 to an empty string will suppress the output of the Customer Provisionable Banner in the response to the **ACT-USER** command.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.
2. **line 2** … line 11 – Line 2 (…Line 11) of the Customer Provisionable Banner.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.

Output format

After receiving the **ENT-BANNER** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

```
;
```

The NE returns the successful completion response even if none of the parameter values are modified using the **ENT-BANNER** command.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the **RTRV-HDR** command.

Detailed behavior description

The command shall update the specified lines of the Customer Provisionable Banner with the text strings provided in the command. If an empty string is specified for a line, then the content of the line is set to an empty string. All lines which are not specified shall remain unchanged.

Banner Display Control

If *line1* is set to an empty string, then the Customer Provisionable Banner is not displayed in the response to the ACT-User command. Setting *line1* to an empty string does not affect *line2 … line11*. The content of *line2 … line11* can still be retrieved with RTRV-BANNER and changed with ENT-BANNER.

Notification behavior

The **ENT-BANNER** command shall generate an ENT-BANNER Notification containing all lines which are specified in the command, independent whether the line has been changed or not.

Example input/output

The following example shows the successful completion of the **ENT-BANNER** command by the NE:

ENT-BANNER:LT-UNITE-123::123456:::line1=″This^is^line^1^of^the^banner″;

```
   LT-UNITE-123 04-12-23 08:01:00
M  123456 COMPLD
;
```

Related TL1 Messages

- “ACT-USER: Activate user” (p. 3-24)
- “RTRV-BANNER” (p. 3-2272)

ENT-BANNER notification

Output format

See definition of REPT DBCHG – Report Database Change.

Command syntax: **ENT-BANNER** :::*spec_block;*

Output parameters

Spec block parameters

1. **line1** – Line 1 of the Customer Provisionable Banner.
   Setting line1 to an empty string will suppress the output of the Customer Provisionable Banner in the response to the **ACT-USER** command.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.
2. **line 2** … line 11 – Line 2 (…Line 11) of the Customer Provisionable Banner.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.

Example output

The example shows a normal notification after an ENT-BANNER command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-01-12,DATE=04-12-23:ENT-BANNER:::line1=\″This is line 1 of the banner\″″
;
```

Error response

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-BANNER** command.

Related TL1 Messages

- “ACT-USER: Activate user” (p. 3-24)
- “RTRV-BANNER” (p. 3-2272)

ENT-CONN: Establish network connections with TNA address parameters

Overview

The **ENT-CONN** command is used to establish one Network Connection between two TNAs in the network. This connection can go through one or multiple network domains.

ENT-CONN creates a Network Connection with specified connection bandwidth between ingress and egress ports connected to the network using TNA addresses on a ONNS controlled NE node.

ENT-CONN shall set up a Network Connection from source *srcTNAAddress*, *srcLogPortID*, *srcLabel* to destination *destTNAAddress*, *destLogPortID*, *destLabel*.

REPT CONNCHG is generated

The **ENT-CONN** command shall generate a **REPT CONNCHG** message.

A REPT PTHCHG notification is generated

The **ENT-CONN** command shall indirectly generate a **REPT PTHCHG** message.

Note

In case setup or deletion of many network connections over several domains are running and reboots of ingress or egress nodes in affected domains occur at the same time, some remain network connection segments may be left over.

Orphan network connection pieces can be identified by applying “RTRV-CONN: Retrieve data relevant for network connections” (p. 3-2294) or “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) commands to affected ingress nodes. Such pieces then can be removed by applying a forced deletion via “DLT-PATH: Delete ONNS path connection” (p. 3-222) in one or all domains affected.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: NA

Abortable: No

Input format

Command syntax:
ENT-CONN:*srcTid:srcAid:ctag:[callID]:servLevel:dir:bandwidth:protType:srcTNAType, srcTNAAddress,srcLogicalPortID,srcLabel:destTNAType,destTNAAddress, destLogicalPortID,destLabel:specBlock;*

Input parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier *srcAid* of the ONNS controller OS command interface port.
   Type: AID.
   Range: snn
   Default: n/a
3. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.
4. **`callID`**
   The *callID* indicates a call of a logical call / connection separation (call capability). This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 .. F).
5. **`servLevel`**
   The service level is mapping the connection type of an ONNS connection segment of the Network Connection. The values of the connection type are representing a mapping of the ONNS parameters *connectionClass*, *disjointness* and *grpSetupType* (for Ethernet connections).
   Type: Numeric string.
   Range: 0..255
6. **`dir`**

Path Direction Topology:

- Unidirectional A-Z (1WAY) or
- Bidirectional (2WAY)

For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
Type: Value set.
Range: 1WAY (default) or 2WAY

7. **`bandwidth`**
   Specifies the bandwidth of the network connection to be setup.
   Type: Alphanumeric value set.
   Range: STS1, STS3, STS12, STS48, STS192.
   Default: None.

8. **`protType`**
   The *protType* defines the requested protection of advertised abstract links (interdomain as well as intradomain).
   protType is the requested setup parameter (via **`ENT-CONN`**) and is only visible at the source node if requested.
   Type: Alphanumeric value set.
   Range:
   - 0x1: only unprotected links shall be considered for routing
   - 1+1: only 1+1 MSP protected links shall be considered for routing)
   - N/A: all links shall be considered for routing, irr. of their protection type

9. **`srcTNAType`**
   srcTNAType specifies the address type of the source TNA address.
   Type: Alphanumeric value set.
   Range:
   - SRCIPV4TNAADDR source: TNA address is a IPv4 address
   - SRCIPV6TNAADDR source: TNA address is a IPv6 address
   - SRCNSAPTNAADDR source: TNA address is a NSAP address

10. **`srcTNAAddress`**
    srcTNAAddress depends on the previous *sourceTNAtype* parameter and is one of:
    - srcIPv4TNAAddress (when *srcTNAtype* = SRCIPV4TNAADDR)
    - srcIPv6TNAAddress (when *srcTNAtype* = SRCIPV6TNAADDR)
    - srcNSAPTNAAddress (when *srcTNAtype* = SRCNSAPTNAADDR)

11. **`srcLogicalPortID`**
    Logical Port Identifier related to the TNA address. *srcLogicalPortID* is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
    Type: numeric value set
    Range: 0 to 4294967295

12. **srcLabel**
    The source label specifies together with the source logical port ID the associated ingress *connPoint* for the network connection to be set up. The *srcLabel* is defined in KLM notation.
    Type: Character string.
    Range:
    Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:
    - S #0..65535
    - U #0..15
    - K #0
    - L #0
    - M #0

    Example for an STS-12c connection service on a OC48 link which starts at timeslot 25: 9-0-0-0-0.
13. **destTNAType**
    destTNAType specifies the address type of the destination TNA address.
    Type: Alphanumeric value set.
    Range:
    - DESTIPV4TNAADDR destination: TNA address is a IPv4 address
    - DESTIPV6TNAADDR destination: TNA address is a IPv6 address
    - DESTNSAPTNAADDR destination: TNA address is a NSAP address
14. **destTNAAddress**
    destTNAAddress depends on the previous *destTNAtype* parameter and is one of:
    - destIPv4TNAAddress (when *destTNAtype* = DESTIPV4TNAADDR)
    - destIPv6TNAAddress (when *destTNAtype* = DESTIPV6TNAADDR)
    - destNSAPTNAAddress (when *destTNAtype* = DESTNSAPTNAADDR)
15. **destLogicalPortID** – Logical Port Identifier related to the TNA address
    srcLogicalPortID is the requested setup parameter (via **ENT-CONN** or UNI) and is only visible on the source node of the network connection.
    Type: Numeric value set.
    Range: 0 to 4294967295.
    Default: None.
16. **destLabel**
    The destination label specifies together with the destination logical port ID the associated egress *connPoint* for the network connection to be set up. The *destLabel* is defined in KLM notation.
    Type: Character string.
    Range:

Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:

- S #0..65535
- U #0..15
- K #0
- L #0
- M #0

Example for an STS-12c connection service on a OC48 link which starts at timeslot 25: 9-0-0-0-0.

Spec Block Parameters

1. **`exclLinkResCls`**
   The exclude list of *linkResCls* is used to exclude interdomain links from the interdomain routing calculation. *exclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
   Type: Alphanumeric value set (4 bytes), every bit defining one link class
   Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
2. **`inclLinkResCls`**
   The include list of *linkResCls* is used to define interdomain links which has to be included for interdomain routing calculation. *inclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
   Type: Alphanumeric value set (4 bytes), every bit defining one link class
   Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF

Output format

Normal system response format to ENT-CONN

If the **`ENT-CONN`** request is accepted by the system, the following normal completion response format below is returned:

```
   srcTid date time
M  ctag COMPLD
   ″CallID=callID″
;
```

Output parameters

Command parameters

1. **srcTid**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **date**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date
3. **time**
   Time. The time parameter is set to the time when the output message is generated. See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time
4. **ctag**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.

5. **`completion code`**
   Standard TL1 completion code result.
   Type: Value set.

   Range:

   - COMPLD (default): for success
   - DENY: in case of errors

6. **`callID`**
   The *callID* is generated by the system, if it is not set by the management sytem in the **`ENT-CONN`** command.
   The *callID* indicates a call of a logical call / connection separation (call capability).
   This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 .. F).

Example input/output

Input/output example for ENT-CONN

The example shows an accepted ENT-CONN command for a bidirectional connection setup:

```
ENT-CONN:LT-NN-
      789:snn:653456:A123B3EF1345A123B3EF1345:0:2WAY:STS12c:0x1:SRCIPV4TNAADDR,
135.13.3.119,1019,9-0-0-0-0:DESTIPV4TNAADDR,135.225.43.128,2,4-0-0-0-0:;


   LT-NN-789 2002-07-18 08:00:00
M  653456 COMPLD
   ″CallID=A123B3EF1345A123B3EF1345″
;
```

Error responses

Validation rule: Input Parameter Source TNA Address not valid

If the value for the input parameter *srcTNAAddress* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid Source TNA Address`

This validation is especially applied in case input parameter *srcTNAAddress* does not address a network TNA address information (that is, does not address a TNA information for the SDH/SONET layer). **`ENT-CONN`** is currently not allowed to be executed on Ethernet Private Line (EPL) layer, that is, with addressing TNA information on corresponding layer.

Validation rule: Input Parameter Destination TNA Address not valid

If the value for the input parameter *destTNAAddress* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid Destination TNA Address`

Validation rule: Input Parameter Source Logical Port ID not valid

If the value for the input parameter *srcLogicalPortID* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid Source LPID`

Validation rule: Labels and bandwidth not matching

If the value for the input parameters *srcLabel*, *destLabel*, and *bandwidth* is not matching, the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, Labels and Bandwidth are not matching`

Note that a value "0-0-0-0-0" for the *srcLabel* is not allowed, while such a value for the *destLabel* is accepted. This validation is thus not be applied in the latter case.

Validation rule: Input Parameter Service Level not valid

If the value for the input parameter *servLevel* is not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid, invalid Service Level`

Validation rule: Inconsistent Tributary Mode of Ingress Port

The following error response shall be returned, if

- for the setup of an adaptive rate network connection the ingress port selected by source TNA address and source logical port information has fixed tributary mode or
- for the setup of a fixed rate network connection the ingress port selected by source TNA address and source logical port information has adaptive tributary mode

Error code:
`SNVS – Status, Not in Valid State, Inconsistent Input Tributary Mode`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
- "DLT-CONN: Delete network connection" (p. 3-139)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT-PTHCHG notification" (p. 3-2149)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)

- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT CONNCHG notification" (p. 3-2104)

ENT-CRS: Enter cross-connection

The ENT-CRS-*modifier* command can be initiated by a user to establish a cross-connection between tributaries in the NE.

The *modifier* in the **ENT-CRS** command indicates the cross-connection rate of the tributary for which the command is applied to. There is no difference if the rate is specified as a SONET rate (STS1, STS3,..) or as SDH-rate (VC3, VC4, …) for the establishment of the cross-connection i.e. the **ENT-CRS**-<*sonet-rate*> yields to the same (with the exception mentioned next) physical and logical behavior as **ENT-CRS**-<*sdh-rate*>. No check is made if the *modifier* matches the interface-standard of the port i.e. if the port is provisioned as SONET (resp. SDH) and the *rate* is a SONET (resp. SDH) rate. The only difference between entering a cross-connection via SONET- or SDH-rate as modifier is that a **RTRV-CRS**-*modifier* will return the same output parameter *rate* as was used when entering the cross-connection.

AU3/AU4 conversion cross-connections are cross-connections between HO an LO tributaries, i.e. between tributaries of different rates. Two different modifiers are defined for conversion cross-connections, one for HO input tributary to LO output tributary and another for LO input tributary to HO output tributary. The input and output tributary rates must fit to the modifier. On the HO side of the conversion cross-connection, STS1 (SONET) and VC3 (SDH) tributaries are considered as equivalent rates.

The number of HO-cross-connections from the same tributary is restricted to 10. This allows 1:N broadcast with N ≤ 10. The number of LO-cross-connections or conversion cross-connections from the same tributary is restricted to two i.e. bridging (1:2) is possible but no broadcast 1:N, N>2.

See section Error response concerning not supported/allowed combinations of input/output for cross-connections.

Physical preconditions prior to establishing a cross-connection

Please note that (for the **ENT-CRS** command processing) it is not necessary that the packs necessary for establishing traffic (i.e. the switch and the optical-packs OP***n*** or GE***n***) are physically present and operable; Instead it is sufficient that those packs are pre-/provisioned and the corresponding logical entities are created in the database of the NE.

Note for ES64U

Interworking between the ES64U and the GE1 Gigabit Ethernet units or electrical I/O port units (EP51 or EP155) is not supported.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 3

Abortable: No

Input format

Command syntax:
ENT-CRS-*modifier*:*tid*:*in_aid*,*out_aid*:*ctag*::[*cct*][,*in_aid2*][:*spec_block*];

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the **ENT-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - AU3TU3
   - TU3AU3.
2. **tid** – Target Identifier
   Target Identifier. Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.

3. **`in_aid`** – Input Access Identifier
   This parameter specifies the source of the cross-connection.
   In case of a 1-way cross-connection the source of a cross-connection is unambiguous. In this case the cross-connection will be retrieved by the RTRV-CRS, which has the identical *in_aid*.
   In case of a 2-way cross-connection the source is not defined (the source could be the *in_aid* or the *out_aid* of the cross-connection). In this case both aids (*in_aid* or *out-aid*) of the cross-connection can be used as the *in_aid* parameter of the **`RTRV-CRS`** command. The **`RTRV-CRS`** command will then return the *in_aid* – parameter used in the **`RTRV-CRS`** command as the *in_aid* of the RTRV-CRS reply.
   Example:
   ENT-CRS-VC4:NEname:1-1-#-#-1-1-1,1-1-#-#-2-1-1:123456:2way;
   RTRV-CRS-VC4:NEname:1-1-#-#-2-1-1:123457;
   M 123457 COMPLD ″1-1-#-#-1-1-,1-1-#-#-2-1-1:vc4….. ;
   Type: GenTribAID or GenLOTribAID.
   Range: No use of the *all* or *usall* or *vall* values within the in_aid is allowed.
4. **`out_aid`** – Output Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   Type: GenTribAID or GenLOTribAID.
   Range: No use of the **`all`** value within the *out_aid* is allowed.
5. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
6. **`cct`** – cross-connection Topology
   This parameter specifies the topology of the cross-connection to be established via the **`ENT-CRS`** command.

   The topology defines

   - the number of legs of the cross-connection
   - whether both legs belong to the same tributary
   - whether a path protection is included.

The following values are possible:

- 1WAY: the cross-connection consists of one leg given by the *in_aid* to the *out_aid*
- 2WAY: the cross-connection consists of two legs given by the *in_aid* to the *out_aid* and the return leg given by the *out_aid* to the *in_aid*
- 1WAYPPROT: the cross-connection consist of one leg (the worker leg) given by *in_aid* to *out_aid* and another leg (the protection leg) given by *in_aid2* to the *out_aid*. The path selector function is associated with the tributary of *out_aid*.

Type: Alphanumeric value string.

Range:

- 1WAY
- 2WAY (default)
- 1WAYPPROT.

7. **`in_aid2`** – Input Access Identifier 2
   This parameter is applicable to path-protected cross-connections only and specifies the source of the protection leg.
   Thus *in_aid* and *out_aid* specify the working leg while *in_aid2* and *out_aid* specify the protection leg of a path-protected cross-connection. For non path-protected cross-connection the parameter *in_aid2* must be omitted.
   Both input aids (*in_aid* and *in_aid2*) have to be related to ports with the same interface mode being either SDH or SONET. This is due to different switch criteria for SONET and SDH.
   Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
   STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
   Range: No use of the *all* or *usall* or *vall* values within the *in_aid2* is allowed.
   Default: None (a valid value is needed on case of a path-protected cross-connection – no default is defined).

Spec block parameters

1. **`htmr`** – Hold Off Timer
   This parameter specifies the value of the hold-off timer for automatic protection switching. This parameter is applicable to path-protected cross-connections only and is omitted for non path-protected cross-connections.
   Type: numeric string [ms]

Range:

- For high order path protection groups, the range is: 0 – 99 ms in 1 ms steps and 100 – 10000 ms in 100 ms steps
- For low order path protection groups, the range is: 0, 25, 50, 75 ms (thus in steps of 25 ms) and 100 – 10000 ms in 100 ms steps.

2. **loca** – Location-a
   This parameter specifies the TID of the node at which service is added to the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections, if the Automatic Squelch Map Calculation function is disabled. The parameter *loca* can be specified for BLSR THROUGH, INTER-BLSR and DROP connections while the Automatic Squelch Map Calculation function is enabled to do a User Override. The *loca* parameter value has to be omitted for ADD, non-BLSR connections.
   A 2way cross-connection is considered to be a DROP if the *out_aid* is a non BLSR high-order tributary.
   For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring node then *loca* is the TID of the node where the related HO circuit is first added to the ring. Refer to the figure “Lower-order cross-connections” (p. 3-2322) for further explanation.
   At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
   Additional information are given for this parameter in the RTRV-CRS section with respect to:
   - figure illustrating BLSR connections,
   - inter-ring cross-connections with collapsed nodes,
   - extra traffic handling.

   Type: TID
   Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.
3. **loca2** – Location-a-2
   This parameter specifies the TID associated with *in_aid2* of the protection leg of a path-protected cross-connection.
   For the syntax, range and default definition please refer to *loca* description.
   Please note that *loca2* parameter is omitted if no *in_aid2* has been specified or for non-BLSR cross-connection.
   A 2way cross-connection is considered to be a DROP if the out_aid is a non BLSR high order tributary. Additional information are given for this parameter in the **RTRV-CRS** command.
   The loca2 parameter value has to be omitted for low-order cross-connections in general.

Currently, path protected LO cross-connections with input tributaries from MS-SPRing/BLSR ports are not supported. As a consequence, the parameter *loca2* needs not to be specified for a LO cross-connection.
Type: TID
Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.

4. **`locz`** – Location-z
   This parameter specifies the TID of the node at which service is dropped from the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections, if the Automatic Squelch Map Calculation function is disabled.
   The parameter *locz* can be specified for BLSR THROUGH, INTER-BLSR and ADD connections while the Automatic Squelch Map Calculation function is enabled to do a User Override. The *locz* parameter value has to be omitted for DROP and non-BLSR connections.
   A 2way cross-connection is considered to be a ADD if the in_aid is a non BLSR high-order tributary.
   For lower-order cross-connections the parameter *locz* is the TID of that ring node in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides. If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring. Refer to the figure "Lower-order cross-connections" (p. 3-2322)for further explanation.
   At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.
   Additional information are given for this parameter in the RTRV-CRS section with respect to:
   - figure illustrating BLSR connections,
   - inter-ring cross-connections with collapsed nodes,
   - extra traffic handling.

   Type: TID
   Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.
5. **`omode`** – Output Mode
   This parameter specifies the output mode of the cross-connection:
   - for a 1-way cross-connection the output is clearly defined by the destination (*out_aid*)
   - for a 2-way cross-connection the output is defined as the source-to-destination direction (*in_aid* to *out_aid*)
   - NORM: this is the normal output i.e. the signal from the *in_aid* is passed through to the *out_aid*;

- IDLE/UNEQ: this mode outputs an unequipped signal;
- AIS: this mode outputs an AIS signal.

omode and adaptive-mode: In case a SONET-port is provisioned to adaptive-mode setting the *omode* to IDLE/UNEQ or AIS will be lead to transmitting the corresponding signal (UNEQ or AIS) on STS1 base for all the STS1 tributary comprising the STS-N signal. Thus for an output of an STS3 cross-connection, if either AIS or IDLE/UNEQ is selected each of the 3 STS1 will output the AIS or IDLE/UNEQ signal. When switching the *omode* back to NORM the original STS3 cross-connection-signal will re-appear again.
omode in case of path-protected cross-connection: In case of a path-protected cross-connection the *omode* affects the output of the path selector and is thus not setable on a per leg base. In other words the *omode* insertion is done after (downstream direction) the selector function.
Type: Alphanumeric value set.

Range:

- NORM (default)
- IDLE/UNEQ
- AIS

6. **`ppgname`** – Path protection group name
This parameter is a user-defined string that identifies a path protection group. This parameter is applicable to path-protected cross-connections only and is omitted for non path-protected cross-connections.
Note: Storage and retrieval of the *ppgname* is case sensitive, but the actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
For details please refer to parameter *ppgname* in "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759).
Default: NULL.

7. **`rme`** – Revertive Mode Enable
This parameter specifies whether the path protection operates in revertive-mode (rme=ENABLE) or non-revertive-mode (rme=DISABLE).
This parameter is applicable to path-protected cross-connections only and is omitted for non path-protected cross-connections.
Type: Alphanumeric value set.

Range:

- ENABLE, DISABLE: in case of *protype* = 1+1
- ENABLE: in case of *protype* = 1xNOPT
- ENABLE is not valid for protocol = 1+1_OPTM

Default:

- DISABLE: in case of protype = 1+1
- ENABLE: in case of protype = 1xNOPT (i.e. protection type 1:1)

The protocol = OPTM can only be choosen in case of protype = 1+1, thus default is DISABLE then.

8. **`rtnomode`** – Return Output Mode
   The parameter *rtnomode* (return output mode) is only applicable for 2way cross-connection and is omitted for 1way and path-protected cross-connections.
   Type: Alphanumeric value set.
   Range: NORM (default), IDLE/UNEQ, AIS

   Note the default values:

   - For a 1way cross-connection the default value is NOT-APPLICABLE, but the parameter must be omitted in the **`ENT-CRS`** command.
   - For a 2way cross-connection the default value is: NORM.
9. **`snctype`** – SNCP Type
   This parameter is used to specify the SNCP mode i.e. the switch criteria used for automatic protection switching. This parameter is mandatory for path-protected cross-connections and only applicable to them. The parameter is omitted for non path-protected cross-connections.
   The parameter specifies the type of the path protection w.r.t. switch criterias used by the protection state machine. The value SNCI (Inherent) is used if server signal fail is used as switch criteria. The value SNCN (Non-Instrusive) is used if server signal fail and defects supervised by a non-intrusive path monitor are used as switch criteria The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
   Type: Alphanumeric value set.
   Range: SNCI, SNCN, NA
10. **`wtr`** – Wait To Restore Time
    This parameter specifies the value of the wait-to-restore time for automatic protection switching. This parameter is applicable to path-protected cross-connections only and is omitted for non path-protected cross-connections.
    Type: numeric string (minute)
    Range: 0 – 60
11. **`xcappl`** – cross-connection application
    This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies. The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.
    The numeric values are assigned to appropriate text strings (see Range below) and the manager should display these value in case of a RTRV-CRS.

    As a possible assignment proposal the following may serve:

    - 0) Unknown
    - 1) 1-Way Point-to-Point
    - 2) 1-Way Path-Protected
    - 3) 1-Way Adjunct Path-Protected
    - 4) 1-Way Conversion Cross-Connection

- 10) 2-Way Point-to-Point
- 11) 2-Way Conversion Cross-Connection
- 20) UPSR (or SNCP Ring) Add, Drop
- 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
- 22) UPSR (or SNCP) Drop, 1-Way Broadcast
- 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
- 30) Logical Ring (or SNCP) Add, Drop
- 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
- 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
- 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
- 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)
- 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
- 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

Type: numerical value set.
Range: {0, 1, 2,…, 255}.
Default: The default value is 0.

12. **`xcnum`** – cross-connection number
This parameter is a 12 digit number, identifying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. described in ″Example input/output″ section of the **`RTRV-CRS`** command like UPSR, DRI etc.). Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in ring interworking scenarios).
Type: numeric value set (12 digit).
Range: 000000000000 .. 999999999999.
Default: The default value is 000000000000.

Output format

If the **`ENT-CRS`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior cct=1WAY, 2WAY

Cross-connections can be established via the ENT-CRS-mod command between tributaries of type:

- STS1-tribs: Tribs related to SDH/SONET ports (i.e. ports of OP***n*** packs) – the AID for these tribs being specified as STS1NumAID (refer to the AID definition) or
- VCG-tribs: Tribs related to VCG-ports (i.e. (logical) ports of GE***n*** pack – please refer to [7], [8], [9, [10] for details) – the AID for these tribs being specified as VCGNumAID (refer to the AID definition).
- PTFs: High-oder path terminations related to asynchronous ports or adjunct ports - the AID for the PTFs are specified as STS1NumAID or TMuxAPTNumAID
- LO tribs: Low-rder tributaries related to HO tributaries of SDH/SONET ports or HO terminations of adjunct ports - the AID for LO tribs are specified as LONumAID or TMuxSLONumAID

Between STS1-tribs, cross-connections can be established for all the specified SDH/SONET rate (i.e. VC3/STS1, VC4/STS3, VC44c/STS12, VC416c/STS48, VC464c/STS192).

AIf a VCG trib, PTF or LO trib is involved in a cross-connection then only the cross-connection rates are possible, which fit to the rate of the VCG trib, PTF or LO trib.

Behavior cct=1WAYPROT

Path-protected cross-connections can be established via the ENT-CRS-mod command between tributaries of type:

- STS1-tribs: Tribs related to SDH/SONET ports – the AID for these tribs being specified as STS1NumAID
- VCG-tribs: Tribs related to VCG-ports – the AID for these tribs being specified as VCGNumAID
- PTFs: High order path terminations related to asynchronous ports or adjunct ports - the AID for the PTFs are specified as STS1NumAID or TMuxAPTNumAID
- LO tribs: Low order tributaries related to HO tributaries of SDH/SONET ports or HO terminations of adjunct ports - the AID for LO tribs are specified as LONumAID or TMuxSLONumAID

For *in_aid* and *in_aid2* only VCG tribs are not allowed.

All path protection group related parameters are only used in the context of ENT-CRS command in order to provide appropriate defaults for different applications (from operator perspective). Retrieval, modification and reporting for these parameters is within entity protection group.

When establishing a cross-connection a entity of type cross-connection is created with the parameter values as

- specified in the **ENT-CRS** command.

Additionally when establishing a path-protected cross-connection a protection entity is created with the parameter values as

- specified in the **ENT-CRS** command,
- (for those parameter not part of ENT-CRS command) with the default-values.

Updated affected entities: General

In case of establishing a path-protected cross-connection the command creates in addition an entity of type protection group (of type PATHDRI) with

- the attributes as entered in the **ENT-CRS** command and
- for those attributes not given by the command by its default values.

This creation is reported by means of an ENT-PROTN-GRP notification – refer to this notification for further details.

In case the *repleg* state is changed by establishing a cross-connection an ED-CRS notification is used to notify the updated *repleg* value.

Updated affected entities: 1-way cross-connection, not resulting in a 2 way cross-connection

In case a 1-way cross-connection is established where no ″equivalent return-leg″ (see below) is already existing one instance of type cross-connection is created with the entered parameter. This is reported by means of an ENT-CRS notification.

An ″equivalent return-leg″ means: when entering a 1-way cross-connection with rate N from in_trib to out_trib a return-leg would be given by

- a 1-way cross-connection from out_trib to in_trib
- with the same rate N
- and in (case of BLSR only) the same *loca* and *locz* values

Example:

ENT-CRS-vc4:NEname:1-1-#-#-3-4-1,1-1-#-#-7-3-1:586::1way:;

REPT DBCHG:

ENT-CRS-VC4:1-1-#-#-3-4-1,1-1-#-#-7-3-1:1WAY,:omode=NORM,xcappl=0, xcnum=000000000000,loca=\″\″,locz=\″\″,repleg=1WAY

Updated affected entities: 1-way cross-connection, resulting in a 2-way cross-connection

In case a 1-way cross-connection is established where an ″equivalent return-leg″ (see below) is already existing one instance of type cross-connection is created with the entered parameter and the repleg-parameter ″2-way″. This is reported by means of an ENT-CRS notification.

Furthermore as stated above the *repleg* status is updated for the already existing leg (i.e. in_aid, out_aid are flipped w.r.t. ENT request) and reported via an ED-CRS notification.

An ″equivalent return-leg″ means: when entering a 1-way cross-connection with rate N from in_trib to out_trib a return-leg would be given by

- a 1-way cross-connection from *out_trib* to *in _trib*
- with the same *rate* N
- and (in case of BLSR only) the same *loca* and *locz* values

Example (Precondition is specified in previous example):

ENT-CRS-vc4:NEname:1-1-#-#-7-3-1,1-1-#-#-3-4-1:587::1way:;

REPT DBCHG:

ENT-CRS-VC4:1-1-#-#-7-3-1,1-1-#-#-3-4-1:1WAY,:omode=NORM,xcappl=0, loca=\″\″,locz=\″\″,repleg=2WAY,rtnxcappl=0,xcnum=000000000000, rtnxcnum=000000000000, rtnomode=NORM

REPT DBCHG:

ED-CRS-VC4:1-1-#-#-3-4-1,1-1-#-#-7-3-1::repleg=2WAY:

Note that in the considered case the ENT-CRS notification for the 2-way cross-connection contains the same information as if it would result directly from an ENT-CRS with the *cct*=2WAY, i.e. the ENT-CRS notification for a *repleg*=2WAY cross-connection is independent from the way it is created (either direct cia *cct*=2WAY or via two separated 1-way cross-connections) and it thus also contains the *rtnxcappl*, *rtnxcnum*, *rtnomode* parameters.

Updated affected entities: 2-way cross-connection

In case a 2-way cross-connection is established one instance of type cross-connection is created with the entered parameter and the *repleg*=2WAY.

This is reported by means of an ENT-CRS notification.

Example:

ENT-CRS-vc4:UNITE-NE:1-1-#-#-3-1-1,1-1-#-#-2-1-7:584::2way:;

REPT DBCHG:

ENT-CRS-VC4:1-1-#-#-3-1-1,1-1-#-#-2-1-7:2WAY,:omode=NORM,xcappl=0, xcnum=000000000000,rtnomode=NORM,loca=\″\″,locz=\″\″,repleg=2WAY,rtnxcappl=0, rtnxcnum=000000000000

Updated affected entities: path-protected cross-connection

In case a path-protected cross-connection is established two instances of type cross-connection is created (one for each leg) with the entered parameters and *repleg*=1WAYPSW and *repleg*=1WAYPSP, respectively.

Furthermore one instance of type protection group is created and notified via an ENT-PROTN-GRP notification.

For details on protection group and the related notifications please refer to the protection group section.

Example (path-protected SDH cross-connection i.e. no adaptive mode for non-BLSR):

ENT-CRS-vc3:NameOfNE:1-1-#-#-1-1-1,1-1-#-#-34-1-1:123456::1WAYPPROT,1-1-#-#-2-1-1:ppgname=\″name1\″,rme=ENABLE,snctype=SNCI,htmr=300,wtr=10;

`REPT DBCHG`:

ENT-CRS-VC3:1-1-#-#-1-1-1,1-1-#-#-34-1-1:1WAYPPROT,1-1-#-#-2-1-1: omode=NORM,xcnum=000000000000,xcappl=0,loca=\″\″,locz=\″\″,repleg=1WAYPSW

`REPT DBCHG`:

ENT-CRS-VC3:1-1-#-#-2-1-1,1-1-#-#-34-1-1:1WAYPPROT,1-1-#-#-1-1-1: omode=NORM,xcnum=000000000000,xcappl=0,loca=\″\″,locz=\″\″,repleg=1WAYPSP

`REPT DBCHG`:

ENT-PROTN-GRP:1-1-#-#-15-1-1::PATHDRI:ppgname=\″name1\″, dripathpn=DEFAULT,htmr=300,protn=1-1-#-#-2-1-1,wkg=1-1-#-#-01-1-1, rme=ENABLE,snctype=SNCI,wtr=10,htmr=300,<CML-BLOCK>

Identification of VCG tributaries:

The system shall allow to create cross-connections at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

ONNS specific behavior - Behavior for *xconcls*

A cross-connection which is created via ENT-CRS shall have implicitly set the *xconcls* parameter to TRADITIONAL, see "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)
.

The only way to explicitly set the *xconcls* parameter is via the **`ED-NNCRS`** command.

ONNS specific behavior - Behavior for tributaries belonging to EDGE ports in a BLSR/MS-SPRING group

When a cross-connection of type *xconcls* = TRADITIONAL is created where a tributary belongs to a port with *portClass* = EDGE, which is part of a BLSR/MS-SPRING group, then any *loca*, *locz* parameter with a value different from the default value is copied from the tributary into the corresponding *loca*, *locz* parameter of the cross-connection.

The *loca*/*locz* parameter in the tributary is reset to its default value. This are notified by means of an ED-STS1 (*istd*=SONET) or ED-VC3 (*istd* = SDH) notification.

Low-order cross-connection specific behavior - Auto-substructuring of HO NIM entities (VC3, STS1)

If the first low-order tributary contained in a dedicated high-order tributary gets connected and the high-order tributary is not yet substructured (substr==NO), then the high-order tributary shall get automatically substructured as follows resulting in the initial setup of the related parameters:

ENT-CRS-PTF::PTF::PTF::

modifier inputsig unequiposig ptfstd

LOVC3 6312 6312 SDH

LOVC12 6312 6312 SDH

LOVT1 2811 2811 SONET

The auto-substructuring is executed on VC4 or STS1 granularity.

Resulting creations of HPT and LONIM entities are notified (by ENT-PTF and ENT-LONIM notifications). The substructure and the related PTF and LONIM objects are bidirectionally created independent from the topology of the cross-connect request.

If a cross-connection at LOVC3 rate causes the auto-substructuring of a HO NIM entity then first the creation of the default substructure (6312) will be reported. After then, an auto-reconfiguration of the PTF substrcuture will take place as described in ID 69853 in order to adapt the affected third to the LOVC3 in the needed direction(s).

Example:

If the following cross-connection is entered: ENT-CRS-LOVC3::1-1-#-#-1-1-1-1,1-1-#-#-2-1-1-22:::1WAY;

Then this causes following notifications:

ED-VC3:1-1-#-#-1-1-1::substr=YES;

ED-VC3:1-1-#-#-2-1-1::substr=YES;

ENT-PTF-HOVC4:1-1-#-#-1-1-1::inputsig=6312,unequiposig=6312,…

ENT-PTF-HOVC4:1-1-#-#-2-1-1::inputsig=6312,unequiposig=6312,…

ENT-LONIM-LOVC:1-1-#-#-1-1-1-1&&-63::pmlpfeses=600,pmlpneses=600,…

ENT-LONIM-LOVC:1-1-#-#-2-1-1-1&&-63::pmlpfeses=600,pmlpneses=600,…

ED-PTF-HOVC4:1-1-#-#-1-1-1::inputsig=13-4212;

ED-PTF-HOVC4:1-1-#-#-2-1-1::unequiposig=2112-13-2112;

ED-LONIM-LOVC:1-1-#-#-1-1-1-1::pmlpfeses=2400,pmlpneses=2400;

Low-order cross-connection specific behavior - Auto-configuration of the HO NIM entity

During the auto-substructuring of a HO NIM as required in ID 69850, the following parameter values are automatically configured:

substr is set to YES.

This are notified with a ED-rr notification.

Low-order cross-connection specific behavior - Auto-configuration of the PTF entity

During the auto-creation of a PTF entity as consequence of the auto-substructuring as required in ID 69850, all parameter is set to default. Although such a requirement is not needed to be stated as this is implicit behavior of entity creations, this has been made since the original single-step-approach had foreseen to copy parameters from the related HO NIM entity. WIth the reduced single-step-approach, this idea has been discarded (probably postponed).

Low-order cross-connection specific behavior - Auto-configuration of LO NIM entities

During the auto-creation of a LONIM entities as consequence of the auto-substructuring as required in ID 69850, all related parameters are set to the default value.

Low-order cross-connection specific behavior - Auto-reconfiguration of the PTF substructure (SDH only)

If the first cross-connection of a third of a substructured VC4 tributary gets connected then a mismatching substructure level is automatically corrected for that third and in the required direction.

The changing HPT parameter inputsig and/or unequiposig is notified (by a ED-PTF notification).

The parameter of the first LONIM entity of that third is set to default values according to the newly represented transmission rate except for the parameters:

neppm, *feppm*, *tcamd* and *ptca*.

Any changed parameters in this context is notified in a single ED-LONIM notification. The remaining 20 other entities of that third of the PTF remain untouched.

Low-order cross-connection specific behavior - Example - Performance monitoring handling

Assuming that a VC4 signal had been initially substructured to 63 times VC12 due to 63 bidirectional LOVC12 cross-connections. For all 63 LONIM entities PM had been enabled for both, far end and near end.

Later the operator removes the first 21 LOVC12 cross-connection and establishes a bidirectional LOVC3 cross-connection instead. Following notifications are to be generated:

ED-PTF-HOVC::1-1-#-#-39-16-1::inputsig=13-4212,unequiposig=13-4212;

ED-LONIM-LOVC::1-1-#-#-39-16-1-1::pmlpfeses=2400,pmlpneses=2400;

Note that performance monitoring commences without interruption using the new default SES threshold values but keeping the previously configured TCA mode and TCA profile (the appropriate rate values are included in the TCA profile). The current bins (for 15 min and 24 hours) are declared as being suspect.

Note also that for LONIM entities addressed by 1-1-#-#-39-16-2&&-20 no notification will be sent as they remain untouched. However, the PM data cannot be retrieved from those entities.

Low-order cross-connection specific behavior - Example - Performance monitoring handling

If low-order signals of a VC4 tributary get connected with different rates (VC3 or VC12) in opposit direction for the same timeslots then the values of inputsig and unequiposig will differ:

1. Initial 1WAY connection may cause both direction to be set to the same value

ENT-CRS-LOVC12::1-1-#-#-1-1-1-1,1-1-#-#-2-1-1-1:::1WAY;

PTF::inputsig[1-1-#-#-1-1-1] = 6312,unequiposig=6312;

PTF::inputsig[1-1-#-#-2-1-1] = 6312,unequiposig=6312;

2. An opposit 1WAY connection may cause one direction to be adapted

ENT-CRS-LOVC3::1-1-#-#-2-1-1-1,1-1-#-#-1-1-1-1:::1WAY;

PTF::inputsig[1-1-#-#-1-1-1] = 6312,unequiposig=13-4212;

PTF::inputsig[1-1-#-#-2-1-1] = 13-4212,unequiposig=6312;

Possibly established performance monitoring between the first and second command for both directions of the LO-VC12 signal (saying for slot 1 and 2 in input direction) would be deleted after the second command for the LONIM 1-1-#-#-2-1-1-1 (thus for slot 2).

Low-order cross-connection specific behavior - Example - Opposit unidirectional cross-connections

If low-order signals of a VC4 tributary get connected with different rates (VC3 or VC12) in opposit direction for the same timeslots then the values of inputsig and unequiposig will differ:

1. Initial 1WAY connection may cause both direction to be set to the same value

ENT-CRS-LOVC12::1-1-#-#-1-1-1-1,1-1-#-#-2-1-1-1:::1WAY;

PTF::inputsig[1-1-#-#-1-1-1] = 6312,unequiposig=6312;

PTF::inputsig[1-1-#-#-2-1-1] = 6312,unequiposig=6312;

2. An opposit 1WAY connection may cause one direction to be adapted

ENT-CRS-LOVC3::1-1-#-#-2-1-1-1,1-1-#-#-1-1-1-1:::1WAY;

PTF::inputsig[1-1-#-#-1-1-1] = 6312,unequiposig=13-4212;

PTF::inputsig[1-1-#-#-2-1-1] = 13-4212,unequiposig=6312;

Possibly established performance monitoring between the first and second command for both directions of the LO-VC12 signal (saying for slot 1 and 2 in input direction) would be deleted after the second command for the LONIM 1-1-#-#-2-1-1-1 (thus for slot 2).

Low-order cross-connection specific behavior - Restricted interface standard for low-order cross-connections

The system shall only accept low-order cross-connection requests if the provided modifier (representing a specific interface standard) matches to the supported interface standard of the whole LOXC pack.

With the reduced single-step-approach, the supported interface standard of the LOXC pack will be determined during activation of its switch capacity. This implies that only one interface standard can be supported at a time.

Conversion cross-connection specific behavior - Auto-substructuring of the HO NIM entity related to the LO side

If the first low-order tributary contained in a dedicated high-order tributary gets connected and the high-order tributary is not yet substructured (*substr*==NO), then the high-order tributary shall get automatically substructured. This results in the creation of a HO PTF with the following parameter values:

- **inputsig**:33
- **unequiposig**:33
- **ptfstd**:SDH

The auto-substructuring is executed on VC4 granularity.

Resulting creations of HPT and 63 LONIM entities are notified (by ENT-PTF and ENT-LONIM notifications). The HPT is created together with all 63 LONIM entities (similar as for LO cross-connections).

Because of the fixed inputsig/unequiposig value 33, only the LONIMs 1, 22 and 43 are usable for conversion cross-connections. Example of the notification behavior caused by the auto-substructuring:

If the following cross-connection is entered:

ENT-CRS-AU3TU3::1-1-#-#-1-1-1,1-1-#-#-2-1-1-22:::1WAY;

Then this causes following notifications:

- ED-STS1:1-1-#-#-1-1-1::substr=CNV;
- ED-VC3:1-1-#-#-2-1-1::substr=CNVSUB;
- ENT-PTF-HOVC4:1-1-#-#-2-1-1::inputsig=33,unequiposig=33,…
- ENT-LONIM-LOVC:1-1-#-#-2-1-1-1&&-63::pmlpfeses=2400,pmlpneses=2400,…

Conversion cross-connection specific behavior - Auto-configuration of the HO NIM entity related to the LO side

During the auto-substructuring of a HO NIM as required in ID 73009, the following parameter values are automatically configured:

- *substr* is set to CNVSUB

This are notified with a ED-VC3 notification.

Conversion cross-connection specific behavior - Auto-configuration of the HO NIM entity related to the HO side

When a higher-order tributary gets connected, then the following parameter values of the HO NIM is automatically configured:

- *substr* is set to CNV

This are notified with a ED-rr notification.

No LO substructure is created at the HO NIM entity related to the HO side of a conversion cross-connection. This means also, that no HO PTF and no LONIMs are created at the HO side of a conversion cross-connection.

Conversion cross-connection specific behavior - Creation of 2-Way conversion cross-connections

In the case of a 2-Way conversion cross-connection, the modifier shall define the type of tributaries usable as input and output tributaries.

A 2-Way conversion cross-connection provides the conversion function in both directions, HO STS1/VC3 to LO VC3 and LO VC3 to HO STS1/VC3. The *modifier* value may imply a specific conversion direction. However, the modifier does only define the tributary types in this case.

Example input/output

Example 1

A non-BLSR 1-way cross-connection with rate vc4 between an OP10G (2.vc4) in slot 01 and an OP2G5 in slot 34 (3.port 3.VC4) will be established by:

```
ENT-CRS-VC4:UNITE-NE:1-1-#-#-1-1-4,1-1-#-#-34-3-7:123456::1WAY;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 2

A 2-way Drop-cross-connection with rate vc44c between an OP10G (2.vc4-4c) in slot10 and an OP10G in slot 17 (1.vc4-4c) will be established – the OP10G in slot10 being in the BLSR.

```
ENT-CRS-STS12:UNITE-NE:1-1-#-#-10-1-13,1-1-#-#-17-1-1:123456::2WAY:LOCA=NE-in-RING;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 3

A SDH path-protected cross-connection consisting of

- a working leg (1-1-#-#-10-1-1 -> 1-1-#-#-34-1-13) and
- a protection leg (1-1-#-#-11-1-1 -> 1-1-#-#-34-1-13) is established. None of the affected tributaries is in an BLSR.

The path protection is specified by

- a name (*ppgname*=″Pathprot.Gr.Name.1″),
- the sncp-type (*snctype* = SNCI),
- the hold-off-timer-value (*htmr*=3000, i.e. 3 sec) and
- the wait-to-restore time (*wtr*=10, i.e. 10min).

```
ENT-CRS-STS12:UNITE-NE:1-1-#-#-10-1-1,1-1-#-#-34-1-13:123456::1WAYPPROT,1-1-#-#-11-1
      -1:ppgname=″Pathprot.Gr.Name.1″,rme=DISABLE,snctype=SNCI,htmr=3000,wtr=10;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 4

A 1-way low-order cross-connection with rate VC3 between an OP10G (3rd VC3 in 2nd VC4) in slot 01 and an OP2G5 in slot 34 (2nd VC3 in 3rd VC4 of port 3) will be established by:

```
ENT-CRS-LOVC3:UNITE-NE:1-1-#-#-10-1-4-43,1-1-#-#-34-3-7-22:\
123456::1WAY;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 5

A 2-way low-order cross-connection with rate VT1.5 between an OP10G (last VT1.5 in the 4th VT group of the 2nd STS1) in slot 01 and an OP2G5 in slot 34 (last VT1.5 in the last VT group in 3rd STS1 of port 3) will be established by:

```
ENT-CRS-LOVT1:UNITE-NE:1-1-#-#-10-1-2-15,1-1-#-#-34-3-3-28:\
123456::2WAY;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Entities must exist for HO cross-connections

Cross-connection command processing verifies that the logical high-order tributaries specified in the high-order cross-connection command do exist.

Error code:

`IENE – Input Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: HO tributary must exist for conversion cross-connection

When creating a conversion cross-connection, the command processing shall verify that the HO STS1/VC3 tributary specified in the command exists.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: No HO cross-connections while LO cross-connections or LO substructure exist

If an attempt is made to create HO cross-connections while LO cross-connections or LO substructure exist at the HO tributaries, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: No conversion-cross-connections while LO cross-connections or LO substructure exist

If an attempt is made to create conversion cross-connections while LO cross-connections or LO substructure exist at the HO tributaries, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections or substructure still exist`

Validation rule: No LO cross-connections while conversion cross-connections exist

If an attempt is made to create LO cross-connections, while a conversion cross-connection exists at the HO tributary or at a LO tributary related to the same HO tributary, which is used for the LO cross-connection, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: Entities must fit LOXC interface standard

Cross-connection command processing shall verify that the specified modifier and the given aid's must fit to the LOXC overall interface standard, i.e. the value of the parameter lostd related to the active LOXC pack.

Error code:

`IDNV – Input, Data Not Valid, Low-order cross-connections of this interface standard not supported`

Validation rule: Conversion cross-connections only with LOXC interface standard SDH

The system shall allow to create conversion cross-connections only if the LOXC interface standard is set to SDH. If the LOXC interface standard is not SDH, then the command is denied.

Error code:

`IDNV – Input, Data Not Valid, Conversion cross-connections not supported with current LO interface standard`

Validation rule: LO cross-connections not at HO tributaries used in HO cross-connections

If the network element receives a ENT-CRS command with low-order tributaries relating to a HO tributary which is involved in a HO cross-connection, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, HO cross-connection exists`

Validation rule: LO tributaries must be creatable for conversion cross-connections

When creating a conversion cross-connection, the command processing shall verify that the low-order tributary specified in the command exists or can be properly created.

A tributary boundary check is included, i.e. only the LO tributary numbers 1, 22, 43 are allowed.

Error code:

`IENE – Input, Entity Not Exists, Specified low-order tributary/ tributaries do not exist/cannot be created (possibly connected at high-order level)`

Validation rule: Entities must be adaptable for LO cross-connections

Cross-connection command processing shall verify that the low-order tributaries specified in the low-order cross-connection command can be properly created by adaptation of the existing substructure. For modifier LOVC3 a tributary boundary check is included, i.e. only the LO tributary numbers 1, 22, 43 are allowed.

Error code:

`IENE – Input, Entity Not Exists, Substructure cannot be adapted`

Examples:

Assuming the following low-order cross-connections have been established:

ENT-CRS-LOVC12::1-1-#-#-1-1-1-22,1-1-#-#-2-1-1-22:::1WAY;

ENT-CRS-LOVC3::1-1-#-#-1-1-1-43,1-1-#-#-2-1-1-43:::1WAY;

The following subsequent cross-connection is invalid:

ENT-CRS-LOVC12::1-1-#-#-1-1-1-44,1-1-#-#-2-1-1-44:::1WAY;

Validation rule: Entities must fit to the interface standard

Cross-connection command processing shall verify that the low-order tributaries specified in the low-order cross-connection command have the same interface standard as the high-order tributary they are contained in.

Error code:

`IENE – Input, Entity Not Exists, Not for this interface standard`

Examples:

Assuming the following low-order cross-connections have been established:

ENT-CRS-VC12::1-1-#-#-1-1-1-22,1-1-#-#-2-1-1-22:::1WAY;

ENT-CRS-VC3::1-1-#-#-1-1-1-43,1-1-#-#-2-1-1-43:::1WAY;

The following subsequent cross-connection is invalid:

ENT-CRS-VT1::1-1-#-#-1-1-1-1,1-1-#-#-2-1-1-1:::1WAY;

Validation rule: Only those parameter present which are needed

Cross-connection command processing verifies according to the given context only those parameter are present which are needed for that context. This means that e.g.

- for entering a 1-way cross-connection the *rtnomode* must not be present
- for entering a non path-protected cross-connection the *ppgname, htmr, wtr, snctype, rme* parameter must not be present.
- for entering a non-BLSR or BLSR-ADD cross-connection the *loca* parameter must not be present
- for entering a non-BLSR or BLSR-DROP cross-connection the *locz* parameter must not be present
- for entering a non-BLSR and unprotected cross-connection the *loca2* parameter must not be present
- for entering a low-order cross-connection the *loca2* and *ppgname* parameters must not be present.

Error code:

`IDNV – Input Data Not Valid, Parameter(s) present which are not allowed`

Validation rule: Both legs must have same rate

Cross-connection command processing verifies that each logical AID specified in the command is consistent with the cross-connection rate. Consistency implies the following condition:

The rate of a new cross-connection from an input tributary that is in use for an existing cross-connection must match the rate of that existing cross-connection.

This can happen if to an existing leg with rate n a further leg is tried to establish with rate $m \neq n$.

Error code:

```
IDNC – Input, Data, Not Consistent, Bridging rate is not compliant with
existing leg
```

Validation rule: AID boundary and rate consistency

Cross-connection command processing shall verify that each logical AID in the command specifying a HO tributary is consistent with the cross-connection rate. Consistency implies the following condition:

- the boundary of the specified tributary is consistent with the specified rate.
- STS-1/VC-3 payloads can start at any position
- the HO side of AU3TU3/TU3AU3 conversion cross-connections can be at any position
- the first (STS-1 equivalent) tributary of an STS-3/VC-4 rate cross-connection must be on a mod3 + 1 boundary (1, 4 , 7, …).
- the first STS-1 of an STS-12/VC-4-4c rate cross-connection must be on a mod12 + 1 boundary (1, 13, 25, …).
- the first STS-1 of an STS-48/VC-4-16c rate cross-connection must be on a mod48 + 1 boundary (1, 49, 97, …).
- the first STS-1 of an STS-192/VC-4-64c rate cross-connection must be on a mod192 + 1 boundary (1, 193, 385, …).

Error code:

```
IDNC – Input, Data, Not Consistent, Tributary boundary is inconsistent
with the indicated rate
```

Validation rule: Tributary provisioning and rate consistency

For fixed rate tributaries, cross-connection command processing verifies that

- if the port identified by logical input AID is provisioned to operate in the fixed signal rate mode, as determined by the tributary mode parameter, then the specified cross-connection rate is equivalent to the provisioned signal rate of the input tributary, as determined by the tributary input signal rate parameter.
- if the port identified by logical output AID is provisioned to operate in the fixed signal rate mode, as determined by the tributary mode parameter, then the specified cross-connection rate is equivalent to the provisioned signal rate of the output tributary, as determined by the tributary unequipped output signal rate parameter. Note that in both cases it does not matter whether the connected tributaries are SONET (STS1/STS3/…) or SDH (VC3/VC4/..) signals with the same transmission rate.

Error code:

```
IDNC – Input, Data, Not Consistent, Specified Tributary does not
support the indicated rate
```

Example:

Assuming an STM-16 (SDH) port named A with 1-1-#-#-1-1 has an input structure of 44-34C.

Assuming an OC-48 (SONET) port named B with 1-1-#-#-2-1 has an output structure of 163.

Both ports (A and B) work in fixed mode (port A since it is SDH, port B since configured).

Following TL1 commands could be sent to connect two tributaries unidirectionally:

- ENT-CRS-VC4::1-1-#-#-1-1-16,1-1-#-#-2-1-48::1WAY:…
- ENT-CRS-STS3::1-1-#-#-1-1-16,1-1-#-#-2-1-48::1WAY:…

Note: In case of adaptive rate no consistency check is required.

Validation rule: Restrictions for LOXC cross-connection rates

For low-order cross-connections, cross-connection command processing shall verify that the rates of the HO tributaries fit to the *modifier* parameter as follows:

- LOVT1 requires STS1 HO tributaries,
- LOVC3 and LOVC12 require VC4 HO tributaries.

Error code:

```
IDNC – Input, Data Not Consistent, Specified Tributary does not support
the indicated rate
```

The existence of an STS1 HO tributary at a SONET port can be verified by checking the *inputsig*, *unequiposig* and *istd* parameters of the related port. Example: If the first HO tributary of a port needs to be successfully checked then the related port parameters should read as follows:

- *inputsig* starting with N1-… (N1 is a number of STS1 signals),
- *unequiposig* starting with N1-… (N1 is a number of STS1 signals),
- *istd* equals SONET.

The existence of a VC4 HO tributary at an SDH port can be verified by checking the *inputsig*, *unequiposig* and *istd* parameters of the related port. Example: If the first HO tributary of a port needs to be successfully checked then the related port parameters should read as follows:

- *inputsig* starting with N4-… (N4 is a number of VC4 signals),
- *unequiposig* starting with N4-… (N4 is a number of VC4 signals),
- *istd* equals SDH.

Validation rule: Pack restrictions for LOXC cross-connection sources

For low-order cross-connections, cross-connection command processing shall verify that any input or output AID shall not reside:

- the OPT2G5 pack,
- the EP51 pack (DS3 and EC1 interfaces),
- the GE1 pack,

- the GE10PL1 pack,
- the ES64U pack

Error code:

`IDNC – Input, Data Not Consistent, LO tributary not supported by that pack`

Validation rule: Pack restrictions for conversion cross-connection sources

For conversion cross-connections, cross-connection command processing shall verify that any input or output AID shall not reside:

- the OPT2G5 pack,
- the GE1 pack,
- the GE10PL1 pack,
- the ES64U pack,
- the TransMUX pack

Error code:

`IDNC – Input, Data Not Consistent, conversion cross-connection not supported by that pack`

Validation rule: Restrictions for LO path protection

If a low-order path protection is created, where the specified input tributaries are already used as input tributaries for another, already existing path protection, then the cross-connection command processing shall verify that:

- the specified working input tributary (in-aid) is also the working input tributary of the existing path protection AND
- the specified protection input tributary (in-aid2) is also the protection input tributary of the existing path protection

Error code: `SNVS – Status, Not in Valid State`

Validation rule: XC pack is (pre-)provisioned

Cross-connection command processing shall verify that at least one XC pack is pre-provisioned or provisioned.

Error code: `SNVS – Status, Not in Valid State, XC Pack not provisioned`

This is also valid for LO cross-connections.

This is also valid for conversion cross-connections.

Validation rule: LOXC pack with more than 0G switch capacity

It is not possible to establish low-order cross-connections if there is no LOXC pack with more than 0 GBit/s low-order switch capacity.

Error code:

`SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Validation rule: LOXC pack with more than 0G switch capacity for conversion cross-connections

It is not possible to establish conversion cross-connections if there is no LOXC pack with more than 0 GBit/s low-order switch capacity.

Error code:

`SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Validation rule: No conversion cross-connections with LOXC/1

When creating conversion cross-connections, the command processing shall verify that the low-order switching pack currently used by the system is not of type LOXC/1.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections not allowed with LOXC/1`

Validation rule: No lower order TransMUX cross-connections when LO switch capacity is lower than 40G

When creating lower-order TransMUX cross-connections, the command processing shall verify that the low-order switching capacity of the LOXC pack in the system is greater or equal to 40 Gbit/s.

Error code:

`SNVS – SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Validation rule: Rate must not exceed half of the port rate (2f-BSLR only)

Cross-connection command processing verifies that in case of a 2f-BLSR that the cross-connection rate does not exceed half the rate of the related port.

Note that this validation rule only applies in case of adaptive mode because for fixed-rate such a payload violation would already been denied in the context of the ED-*rr* command.

Error code:

`IDNC – Input, Data, Not Consistent, 2f-BSLR rate exceeds half of the port rate`

Validation rule: path-protected cross-connection has unique input AIDs

Cross-connection command processing verifies that for path-protected cross-connections the two input tributaries are unique i.e. the in_aid has to be different from in_aid2.

Error code:

`IDNV – Input Data Not Vaild, Both input tributaries are identical`

Validation rule: path-protected cross-connection has inputs from ports with same interface standards

Cross-connection command processing denies requests to establish path-protected cross-connections where the ports of the logical input tributaries are not provisioned for the same standard, either SONET or SDH – this is due to different switching criteria.

Error code:

`SNVS – Status, Not in Valid State, Different interface standards for input tributaries`

Validation rule: No path-protected conversion cross-connection

Cross-connection command processing shall deny requests to establish pathprotected conversion cross-connections, i.e. *modifier* = AU3TU3 or TU3AU3.

Error code:

`IDNC – Input, Data Not Consistent, pathprotected conversion cross-connections not allowed`

Validation rule: No trib of any MSP port as *in_aid* for LO path-protected cross-connection

Cross-connection command processing shall verify that for lower order pathprotected cross-connections no tributary is used as *in_aid* or *in_aid2* where the related port is a member of a 1+1 or 1:1 multiplex section/line protection group.

Error code:

`SNVS – Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO crossconnection`

Validation rule: No trib of any MSP 1x1 protection port as *in_aid* for HO pathprotected cross-connection

Cross-connection command processing shall verify that for higher order pathprotected cross-connections no tributary is used as *in_aid* or *in_aid2* where the related port is used as protection port of a 1:1 multiplex section/line protection group.

Error code:

`SNVS – Status, Not in Valid State, trib of 1x1 protection port as input-trib in pathprotected crossconnection`

Validation rule: No tributary of an MS-SPRing/BLSR port for conversion cross-connection

Cross connection command processing shall verify that for conversion cross-connections no tributary at an MS-SPRing/BLSR port is used for the HO or LO side of a conversion cross-connection.

Error code:
`IDNC – Input, Data Not Consistent, No trib of MS-SPRing/BLSR port allowed for conversion cross-connection`

Validation rule: No multiple added XCs for MS-SPRing TOP

Cross-connection command processing shall verify that for pathprotected cross-connections the output tributary and one of the two input tributaries are not a worker tributary of an port being a member of a BLSR/MS-SPRing protection group operating in TOP mode (i.e. protnmode = ’SHORTENED’ or ’TOPEX’).

Error code:
`SNVS – Status, Not in Valid State, multiple added XCs for MS-SPRing TOP`

Validation rule: Path-protected crossconnection: missing parameters

Cross-connection command processing shall deny requests to establish path-protected cross-connections if one of the mandatoy parameters *htmr*, *rme*, *snctype* or *wtr* is missing.

Error code: `IPMS -Input, Parameter MiSsing, <parameter name> is missing`

Validation rule: No Spare capacity for HO cross-connection legs on LOXC

Cross-connection command processing shall deny requests to establish low-order cross-connections or conversion cross-connections if a newly to be substructured HO tributary cannot be connected to the LO switch matrix .

Error code: `SARB – Status, All Resources Busy, LOXC capacity exhausted`

Validation rule: No cross-connection loopback existing

Cross-connection command processing denies commands requesting a cross-connection to a tributary with an existing cross-connection loopback.

Error code:
`SNVS – Status, Not in Valid State, Tributary has existing loopback`

Note: There is no command validation concerning facility loopback i.e. the cross-connection establishing is possible even with a facility loopback present.

Validation rule: No test access session

Cross-connection command processing shall deny commands requesting a cross-connection to a tributary which is used in a test access session either as tributary under test or as test access tributary. The denial of the cross-connection request is independent whether the direction of the tributary (input or output) is used in the test access session or not.

Error code:
`SNVS – Status, Not in Valid State, Tributary used in test access session`

With this validation rule it is not allowed to add a 1-way cross-connection at a tributary where already another 1-way cross-connection exists in the other direction. This shall avoid cross-connection topology conversions.

Validation rule: Restrict number of cross-connection using same input tributary

Cross-connection command processing verifies that the number of cross-connections originating from the same tributary is restricted as defined for the specific release.

Error code:

`SNVS – Status, Not in Valid State, Too many cross-connection originating from same tributary`

Validation rule: TIDs consistent with existing cross-connection

Cross-connection command processing verifies that for each leg of the cross-connection for which the logical input AID is used in any pre-existing cross-connection that the source TID is identical to that associated with the pre-existing cross-connection. This hold for BLSR cross-connections where both legs of a bridge must originate from the same NE.

Error code:

`SNVS – Status, Not in Valid State, TID inconsistent with existing cross-connection`

Validation rule: Output not in use

Cross-connection command processing verifies that logical output AID is not in use as a destination for an existing cross-connection.

Error code:

`SNVS – Status, Not in Valid State, Output is already crossconnected`

Validation rule: 2 Fiber BLSR/MS-SPRing protection access (Extra Traffic)

Cross-connection command processing shall allow commands requesting higher order or lower order cross-connections to the protection tributaries of a port being a member of a 2 Fiber BLSR/MS-SPRing protection group only if the port rate is OC48/STM16 or OC192/STM64.

Error code:

`SNVS – Status, Not in Valid State, Command refers to protection tributaries of a BLSR/MS-SPRing protection`

Validation rule: 4 Fiber BLSR/MS-SPRing protection access (Extra Traffic)

Cross-connection command processing shall allow commands requesting higher order or lower order cross-connections to the protection tributaries of a port being a member of a 4 Fiber BLSR/MS-SPRing protection group only if the related protection group has the parameter *protnmode* set to TOPEX.

Error code:

`SNVS – Status, Not in Valid State, Command refers to protection tributaries of a BLSR/MS-SPRing protection`

Validation rule: No connections between BLSR/MS-SPRing working and protection tribs

Cross-connection command processing shall deny commands requesting higher order or lower order cross-connections between working tributaries and the protection tributaries which belong to the same or different BLSR/MS-SPRing protection group.

Error code:

`IDNC – Input, Data Not Consistent, Connections between BLSR/MS-SPRing working and protection tributaries are not allowed`

Validation rule: No low-order path protection with input tributaries from BLSR/MS-SPRing ports

Cross-connection command processing shall deny commands requesting low-order path-protected cross-connections having at least one input tributary at a BLSR/MS-SPRing port.

Error code:

`IDNC – Input, Data Not Consistent, No low-order path protection with input tributaries from BLSR/MS-SPRing ports`

Validation rule: No low-order path protection with input tributaries from the protection bandwidth of 2-fiber MS-SPRing/BLSR protection

Cross-connection command processing shall deny commands requesting low-order path protected cross-connections having at least one input tributary (in_aid or in_aid2) within the protection bandwidth of 2-fiber MS-SPRing/BLSR protection.

Error code:

`IDNC – Input, Data Not Consistent, No low-order path protection with input tributaries from protection bandwidth of 2-fiber MS-SPRing/BLSR protection`

Validation rule: Corresponding *loca*/*loca2*/*locz* information is missing

Cross-connection command processing shall deny commands requesting cross-connections to worker tributaries on ports in BLSR/MS-SPRing groups, which have Automatic Squelch Map Calculation disabled, if corresponding loca/loca2/locz information is missing in the command.

Error code:

`IDNC – Input, Data Not Consistent, Corresponding loca/locz information is missing`

Validation rule: Consistent *loca*/*loca2*/*locz* for all low-order cross-connections using the same HO PTF at an MS-SPRing port

Cross-connection command processing shall deny the creation of a lower-order cross-connection at an MS-SPRing port, if the *loca*/*loca2*/*locz* information specified in the command does not match the *loca*/*loca2*/*locz* values of already existing low-order cross-connections using the same HO PTF at the MS-SPRing port or if both input tributaries of a lower-order path protected cross-connection use the same HO PTF and *loca*/*loca2* are specified in the command with different values.

Error code:

`IDNC – Input, Data Not Consistent, Inconsistent loca/locz information`

Validation rule: Consistent loca/locz for all conversion cross-connections using the same HO PTF at an MS-SPRing port

Cross-connection command processing shall deny commands, if a conversion cross-connection is created at an MS-SPRing port, and the *loca/locz* information specified in the command does not match the *loca/locz* values of already existing conversion cross-connections using the same HO PTF at the MS-SPRing port.

Error code:

`IDNC – Input, Data Not Consistent, Inconsistent loca/locz information`

Validation rule: loca/locz information only retrievable for EXTRA TRAFFIC

Cross-connection command processing shall deny commands, if affected tributaries belong to EXTRA TRAFFIC and *loca*/*locz* information is specified in the command. The *loca*/*locz* information is automatically created for EXTRA TRAFFIC by the NE.

Error code:

```
SNVS – Status, Not in Valid State, Command refers to protection
tributaries of a BLSR/MS-SPRing protection
```

Validation rule: SNC Type Compliance

Cross-connection command processing verifies the following when establishing a higher-order path-protected cross-connection:

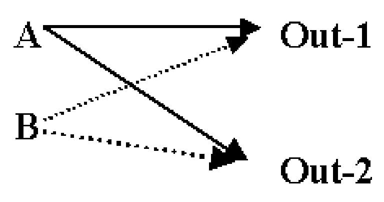
If there are any existing HO path protection groups that are *co-located* to the HO cross-connection to be established, then the SNC protection type parameter in the **`ENT-CRS`** command has the same value as the SNC protection type attribute of all of the *co-located* path protection groups.

A path protection group P is defined for this requirement to be *co-located* to a cross-connection C if any of the logical input tributaries of the cross-connections belonging to P are logical input tributaries of C – see the example below.

Error code:

```
IDNC – Input, Data, Not Consistent, SNC Type not compliant with
existing path-protected cross-connection
```

As an example consider the figure below: A path-protected cross-connection A, B $\rightarrow$ Out-1 may be established with the SNC-Type of e.g. SNCI; If then establishing another path-protected cross-connection A, B $\rightarrow$ Out-2, which has a common tributary (here tributary A) with the former path-protected cross-connection, the SNC-Type of this new one has to be SNCI as well.



Legend: see Fig in the RTRV-CRS section

Validation rule: No VCG-trib as *in_aid* (for path-protected cross-connection)

Cross-connection command processing shall verify that for path-protected cross-connection no VCG-trib is used as *in_aid* or *in_aid2*.

Error code:

`IDNC – Input Data Not Consistent, VGC-trib as input-trib in pathprotected crosconnection`

Validation rule: No OPT2G5 related trib as *in_aid* (for pathprotected cross-connection)

Cross-connection command processing verifies that for path-protected cross-connection no tributary from OPT2G5 is used as *in_aid* or *in_aid2*. The case of path-protected cross-connection in conjunction with OPT2G5 only makes sense for both input-tribs *(in_aid* and *in_aid2*) being non-OPT2G5 STS-tribs and the *out_aid* being a tributary on the OPT2G5 pack..

Error code:

`IDNC – Input, Data Not Consistent, no OPT2G5-trib as input-trib in pathprotected crossconnection`

Validation rule: Maximum number of *ppgnames* exceeded

Cross-connection command processing verifies that for path-protected cross-connection the maximum number of ppgnames is not exceeded if a new (not already used *as ppgname* for another protection group) is specified.

Error code:

`SSRE – Status, System Resources Exceeded, limit of ppgnames exceeded`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* INNI

Cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* INNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass INNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of *portClass* INNI

For low-order cross-connections, cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* INNI.

Error code:

`IDNC – Input, Data Not Consistent, Tributaries must not be on ports of portClass INNI`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* UNI

Cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* UNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass UNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of *portClass* UNI

For low-order cross-connections, cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* UNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass UNI`

Validation rule (ONNS specific): Tributaries must not belong to ports of *portClass* ENNI

Cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* ENNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI`

Validation rule (ONNS specific): LO Tributaries must not belong to ports of ***portClass*** ENNI

For low-order cross-connections, cross-connection command processing shall verify, that all affected input and output tributaries do not belong to ports of *portClass* ENNI.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI`

Validation rule (ONNS specific): Restrict number of crossconnection using same input tributary on EDGE port

Cross-connection command processing shall verify that the number of cross-connections originating from the same tributary is restricted to two, if the affected port is of port class EDGE.

Error code:

`SNVS – Status, Not in Valid State, Port Class must not be EDGE or INNI`

Validation rule (ONNS specific): TRADITIONAL and NN cross-connections not on the same tributary

Cross connection command processing shall verify, that there is no cross-connection of xconcls NN at any of the tributaries specified in the command. The validation rule is only applied for tributaries at EDGE ports which are specified in the command.

Error code:

`SNVS – Status, Not in Valid State, Cross Connection of xconcls NN exists`

Related TL1 messages

- "DLT-CRS: Delete cross-connection" (p. 3-145)
- "ED-CRS: Edit cross-connection" (p. 3-292)
- "ENT-ROLL: Enter roll" (p. 3-1796)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)

References

GR-199-CORE

ENT-CRS notification

Output format

Syntax: **ENT-CRS**-*modifier*:*in_aid*,*out_aid*:*cct*,[*in_aid2*]:*spec-block*;

Output parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the **ENT-CRS** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command.
   Type: Alphanumeric value set.
   Range:
   - LOVC12
   - LOVC3
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - LOVT1
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - AU3TU3
   - TU3AU3
2. **in_aid** – Input Access Identifier
   The parameter *in_aid* (Input Access Identifier) specifies the tributary which is the source of the cross-connection.
   In case of a 1way cross-connection this parameter specifies the source of the cross-connection.
   In case of a 2way cross-connection this parameter is identical to the one used as input parameter (please refer also to the description of the input parameter *in_aid* of the **RTRV-CRS** command).
   In case that a wildcard AID has been used to retrieve 2way cross-connections, then either of the input/output AIDs of the cross-connections will be returned as *in_aid*

in the response. For conversion cross-connections the returned *in_aid* is independent, whether the wildcard AID in RTRV-CRS specifies a HO or LO tributary.
In case of a path-protected cross-connection this parameter may specify either the source of the working leg or the source of the protection leg – the distinction is made via the spec-block parameter *repleg* (see below).
If *in_aid* specifies the working-leg part than *in_aid2* (see below) specifies the protection-leg part and vice versa.
Type: GenTribAID or GenLOTribAID. STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs, VCGNumAID are used for high-order tributaries on e.g. GE***n***, OPT2G5 packs, GenLOTribAID are used for low-order tributaries on I/O packs e.g. OP***n***, EP155.
Range: (**`all`** values shall never occur as part of an output parameter)

3. **`out_aid`** – Output Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   Type: GenTribAID or GenLOTribAID.
   Range: No use of the **`all`** or **`usall`** or **`vall`** values within the *out_aid* is allowed.
4. **`cct`** – cross-connection Topology
   This parameter specifies the topology of the cross-connection to be established via the **`ENT-CRS`** command.

   The topology defines

   - – the number of legs of the cross-connection
   - – whether both legs belong to the same tributary
   - – whether a path protection is included.

   The following values are possible:

   - 1WAY: the cross-connection consists of one leg given by the *in_aid* to the *out_aid*
   - 2WAY: the cross-connection consists of two legs given by the *in_aid* to the *out_aid* and the return leg given by the *out_aid* to the *in_aid*
   - 1WAYPPROT: the cross-connection consist of one leg (the worker leg) given by *in_aid* to *out_aid* and another leg (the protection leg) given by *in_aid2* to the *out_aid*. The path selector function is associated with the tributary of *out_aid*.

   Type: Alphanumeric value string.

   Range:

   - 1WAY
   - 2WAY (default)
   - 1WAYPPROT.
5. **`in_aid2`** – Second Input Access Identifier
   This parameter is applicable to and is present for path-protected cross-connections only and specifies either the source of the protection leg or the source of the working leg.

This depends on the value of the *repleg* parameter:

- E.g. for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSW the worker leg is given by IN1 -> OUT and the protection leg is given by IN2 -> OUT;
- for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSP the worker leg is given by IN2 -> OUT and the protection leg is given by IN1 -> OUT.

Thus *in_aid* and *out_aid* specify one leg of the legpair of a path-protected cross-connection, while *in_aid2* and *out_aid* specify the second leg.
Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
Note: Using VCGNumAID (when using e.g. GE***n*** packs) is NOT applicable as those packs can not be the input to a path-protected cross-connection. Therefore specific AID types are mentioned instead of GenTribAID.
Range: n.a. (**`all`** and **`usall`** values shall never occur as part of an output parameter).

Spec block parameters

1. **`loca`** – Location-a

   This parameter specifies the TID of the node at which service is added to the BLSR.
   This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections.
   The *loca* parameter value is an empty string for ADD and non-BLSR connections.
   A 2way cross-connection is considered to be a DROP if the out_aid is a non BLSR tributary.
   For description of a 2way scenario see RTRV-CRS command.
   Inter Ring Cross-connections with collapsed node: Note that beside the above described THROUGH case there can be also another situation where both *loca* and *locz* are different from the empty string: If two rings are interconnected in a collapsed node then *loca* refers to the (add-node of the) ring where the signal is dropped from, and *locz* refers to the (drop-node of the) ring where the signal is added to. In such cases the cross-connection cannot be provisioned as bidirectional, thus the information is still unambiguous.
   A note to the loca2 parameter: Although *loca2* (Location-a-2) is a parameter which is used in the **`ENT-CRS`** command, *loca2* is not retrieved explicitly. This is due to the fact that for path-protected cross-connections – and only for those this *loca2* is applicable – the retrieval will be on a leg-basis and not on a leg-pair basis. In other words: If a path-protected cross-connection is retrieved two entries will be sent, one for the working leg (incl. the *loca* and *locz* parameter) and one for the protection leg (incl. the *loca* parameter and the same *locz* parameter, and in this case the *loca* returned is the *loca2* as entered when establishing the path-protected cross-connection).

The parameter *loca* is also required for EXTRA TRAFFIC connections. (In contrast to incoming references from other commands to this whole chapter it is exclusively applicable for the command **RTRV-CRS**). For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of *loca* refer to the parameter *sqfa*.
For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring node then loca is the TID of the node where the related HO circuit is first added to the ring. The picture "Lower-order cross-connections" (p. 3-2322)shows an example of a LO path through a part of a ring. The LO path is added to/dropped from the ring in NE-1 and NE-2. 2-way cross-connections are considered to be either an Add or a Drop connection depending whether the input or the output tributary is at the ring port. In NE-3 there is a HO through connection, which contains the LO path. In NE-3 there is no HO path termination. In NE-4 there is a LO through connection. Consequently, there are HO path terminations in NE-4.

The fact that loca refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same loca value.
- Any LO through-connection will be treated as add/drop. loca will identify the node of the through connection, and not the node where the LO service is added to the ring.

At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
Type: TID or empty-string (default)
Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.

2. **locz** – Location-z
This parameter *locz* (Location-z) specifies the TID of the node at which service is dropped from the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections. The *locz* parameter value has to be omitted for DROP and non-BLSR connections.
A 2way cross-connection is considered to be a ADD if the in_aid is a non BLSR tributary.
Refer to figure "2-way scenario" (p. 3-2322)in the RTRV-CRS section for further explanation.
locz is also required for EXTRA TRAFFIC connections.
For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of locz refer to the parameter *sqfz*.
For lower-order cross-connections the parameter *locz* is the TID of that ring node

in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides. If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring. Refer to figure "Lower-order cross-connections" (p. 3-2322) in the RTRV-CRS section for further explanation.

The fact that locz refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same *locz* value.
- Any LO through-connection will be treated as add/drop. *locz* will identify the node of the through connection, and not the node where the LO service is dropped from the ring.

At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.
Type: TID or empty-string (default)
Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.

3. **`omode`** – Output Mode

   This parameter specifies the output mode of the cross-connection:

   - for a 1-way cross-connection the output is clearly defined by the destination (*out_aid*)
   - for a 2-way cross-connection the output is defined as the source-to-destination direction (*in_aid* to *out_aid*)
   - NORM: this is the normal output i.e. the signal from the *in_aid* is passed through to the *out_aid*;
   - IDLE/UNEQ: this mode outputs an unequipped signal;
   - AIS: this mode outputs an AIS signal.

   Type: Alphanumeric value set.

   Range:

   - NORM (default)
   - IDLE/UNEQ
   - AIS

4. **`onnsConnControl`** – ONNS Connection Control
   This parameter indicates for cross-connections of *xconcls* NN, whether the ONNS connection a cross-connection is assigned to is controlled by ONNS directly or by an external control (setup via UNI or eNNI or as SPC connection).
   For cross-connections of *xconcls* TRADITIONAL the value UNKNOWN is reported for this parameter.
   The value UNKNOWN shall also be reported for cross-connections of *xconcls* NN, the onnsConnControl of which cannot be determined.
   Type: Alphanumeric value set.

Range:

- UNKNOWN
- ONNS
- EXT

5. **`onnsPathID`** – ONNS Path Identity
   This parameter indicates for cross-connections of *xconcls* NN the path ID of the ONNS path a cross-connection is assigned to.
   For cross-connections of *xconcls* TRADITIONAL an empty string is reported for this parameter.
   Type: Character string of length up to 20 or empty string.
   Range:
   nodeID-Index
   - nodeID: four dot separated numeric strings: each string 0..255
   - Index: Numeric string 0-65535

   Example: 29.13.196.203-65111
   Default: empty string
6. **`repleg`** – Reported Leg Pair
   This parameter provides compact information about the leg(s) of cross-connections and is derived, as shown in the table below, from a combination of the following elementary parameter:
   - atomic cross-connection topology
   - leg status
   - path-protected group working leg (if applicable i.e. being part of a protection group).

Atomic cross-connection Topology	Path Protection Leg Role	Leg Status	Reported Legpair (*repleg*)
point-to-point	n.a.	1-way	1WAY
point-to-point	n.a.	2-way	2WAY
path-protected	W	1-way	1WAYPSW
path-protected	P	1-way	1WAYPSP

Atomic cross-connection topology:

Two atomic cross-connection topologies are supported:

- Point to point: The point to point cross-connection consists of one leg only, which is defined by its source-trib (*in_aid*) and its destination-trib (*out_aid*) and its particular transmission rate.
- path-protected: The path-protected cross-connection consists of a path protection group with two legs, a working leg and a protection leg – the declaration of working respectively protection is made when establishing the cross-connection (ENT-CRS command).

The two legs are defined by two input tributaries and one output tributary – thus one input tributary and the output tributary form the working leg while the other input tributary and the output tributary form the protection leg.

Note that:

both input tributaries must belong to ports with the same interface standard (either SONET or SDH), due to different switch criteria;

both legs must be at the same cross-connection rate.

Leg status:

The leg status indicates whether a leg is monitored and reported as a 1-way cross-connection or a leg pair i.e. a 2-way cross-connection.

1-way: The leg status for a path-protected cross-connection is always defined as 1-way.

2-way: The leg status for a (non path-protected) cross-connection is set to 2-way if all the following conditions are fulfilled: there are 2 cross-connections in opposite direction between the same tributaries; both of these cross-connections have the same rate (independent if the SONET or SDH rate was used when establishing i.e. VC3 is equivalent to STS1, VC4 is equivalent to STS3 etc); the source and destination TIDs do match i.e. the source TID of one leg is the destination TID of the other leg and vice versa (this condition of course holds only in case of BLSR cross-connections).

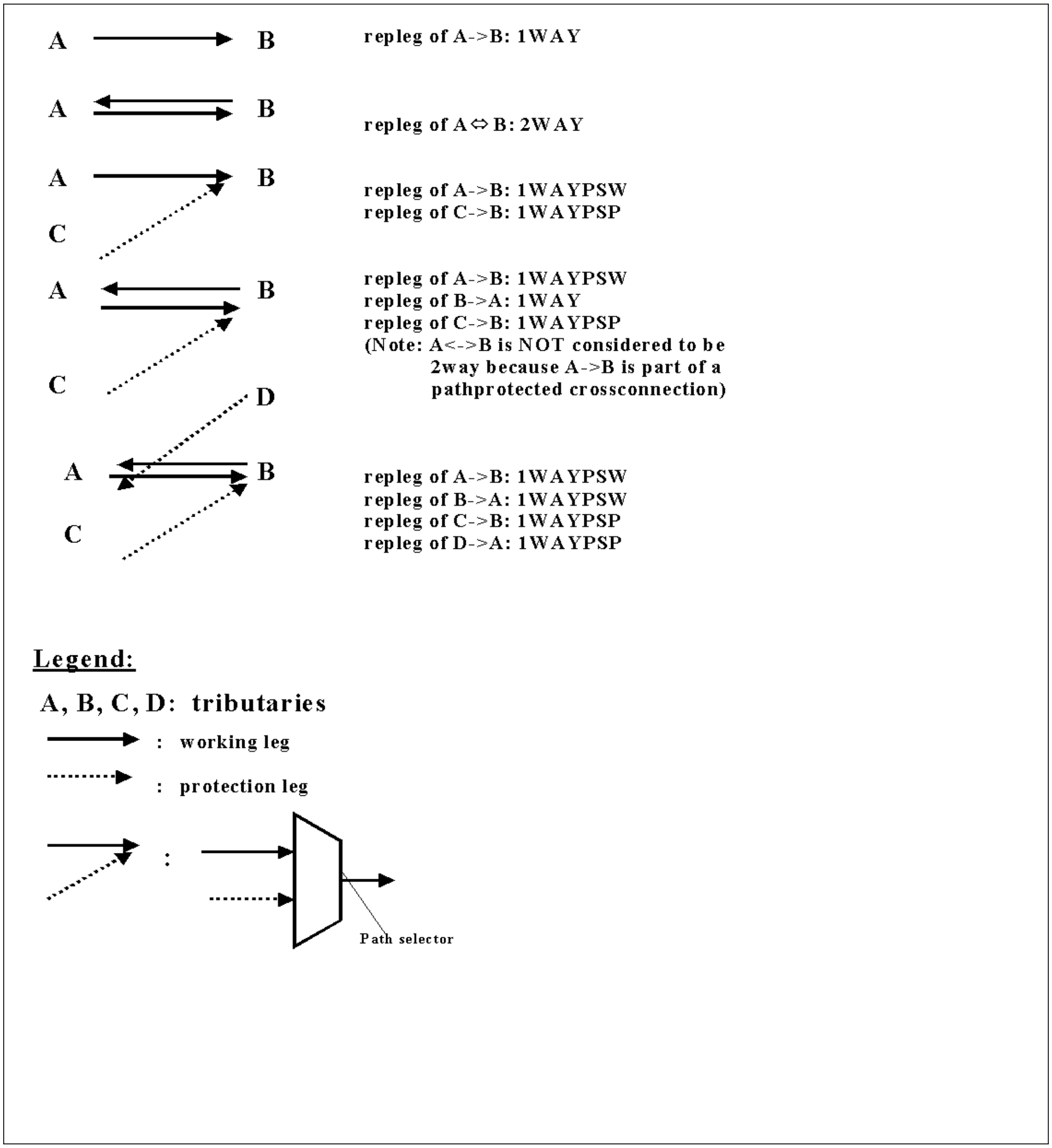
Note that the criteria for a 2-way leg-pair do not depend on whether the two legs were set up at the same time, nor on whether either leg was set up as part of a 2-way cross-connection topology

Path Protection leg role:

This indicates if the leg of a path protection group is defined as the working (W) leg or the protection (P) leg.

1WAYPSW and 1WAYPSP stands for 1-way path switched working and 1-way path switched protection respectively.

The following figure shows examples for the *repleg* parameter for several cross-connection configurations including compound cross-connection.



Range:

- 1WAY
- 2WAY
- 1WAYPSW
- 1WAYPSP.

The values 1WAYPAP and 1WAYPAW are not supported in the initial release.

7. **`rtnomode`** – Return Output Mode
   The parameter *rtnomode* (return output mode) is only applicable for 2way cross-connection and is omitted for 1way and path-protected cross-connections.
   Type: Alphanumeric value set.
   Range: NORM (default), IDLE/UNEQ, AIS
8. **`rtnxcappl`** – Return cross-connection application
   This parameter is a numeric value which indicates the application for which the return cross-connection is used in the context of compound cross-connection topologies.
   - for a 1-way cross-connection this parameter is not applicable and will thus be omitted.
   - for a 2-way cross-connection the parameter is present.

   Type: same as *xcappl*.
   Range: same as *xcappl*.
   Default: The default value is 0.
9. **`rtnxcnum`** – Return cross-connection number
   This parameter is a multiple-digit number identifying the cross-connection number for the return direction (destination to source i.e. *out_aid* to *in_aid*) of a 2-way cross-connection.
   - for a 1-way cross-connection this parameter is not applicable and will thus be omitted.
   - for a 2-way cross-connection the parameter is present.

   Type: same as *xcnum*.
   Range: same as *xcnum*.
   Default: The default value is 000000000000.
10. **`xcappl`** – cross-connection application
    This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies.
    The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.
    The numeric values are assigned to appropriate text strings (see Range below) and the manager should display these value in case of a "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).

    As a possible assignment proposal the following may serve:
    - 0) Unknown
    - 1) 1-Way Point-to-Point
    - 2) 1-Way Path-Protected
    - 3) 1-Way Adjunct Path-Protected
    - 4) 1-Way Conversion Cross-Connection
    - 10) 2-Way Point-to-Point
    - 11) 2-Way Conversion Cross-Connection
    - 20) UPSR (or SNCP Ring) Add, Drop

- 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
- 22) UPSR (or SNCP) Drop, 1-Way Broadcast
- 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
- 30) Logical Ring (or SNCP) Add, Drop
- 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
- 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
- 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
- 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)
- 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
- 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

Type: numerical value set.
Range: {0, 1, 2,…, 255}.
Default: The default value is 0.

11. **`xcnum`** – cross-connection number
This parameter is a 12 digit number, identifying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. described in ″Example input/output″ section of the **`RTRV-CRS`** command like UPSR, DRI etc.).
Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in ring interworking scenarios).
Type: numeric value set (12 digit).
Two digits for the bay, one digit for the shelf, two digits for the slot, two digit for the port, three digits for the high-order tributary, and two digits for the low-order tributary.
Range: 000000000000 .. 999999999999.
Default: The default value is 000000000000.

12. **`xconcls`** – cross-connection class
This parameter identifies, whether the cross-connection has been setup in the TRADITIONAL way by an operator via the **`ENT-CRS`** command or via the NN application as part of an ENT-PATH command. This parameter therefore also indicates who is the owner of the cross-connection and can modify or delete it.
Type: Alphanumeric value set.
Range: TRADITIONAL, NN.
For low order cross-connections, the value range is limited to TRADITIONAL.

Example output

```
LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
```

```
   ”TIME=08-00-12,DATE=00-01-01:ENT-CRS-VC4:1-1-#-#-1-1-1,\
    1-1-#-#-2-1-1:2WAY:lpbkstat=NO,omode=NORM,repleg=2WAY,\
    rtnomode=NORM,rtnxcappl=20,rtnxcnum=123456789012,\
    xconcls=TRADITIONAL,xcappl=20,xcnum=123456789012”
;
```

ENT-CRS-FECOM: Enter transparent DCC cross-connection

The **ENT-CRS-FECOM** command can be initiated by the users to establish a transparent DCC cross-connection.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 3

Abortable: No.

Input format

Command syntax: **ENT-CRS-FECOM**:*tid:port_aid1,port_aid2,dcctype:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **port_aid1** – DCC port 1
   This parameter indicates one (first) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID
   Range: SPortAID. The keyword **all** is allowed not.
4. **port_aid2** – DCC port 2
   This parameter indicates one (second) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID.
   Range: SPortAID. The keyword **all** is allowed not.
5. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS transparent DCC cross-connection parameters of the OP***n*** port is reported by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.

Output format

In response to a valid ENT-CRS-FECOM command, the following output is returned:

```
    sid date time
 M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **ENT-CRS-FECOM** command establishes a transparent DCC cross-connection. The transparent DCC cross-connection will be established between *port_aid1* and *port_aid2* for the specified dcctype. The System supports only bidirectional transparent DCC cross-connections.

Update affected entities

The established bidirectional transparent DCC cross-connection will be reported via ENT-CRS-FECOM notification reporting the established transparent DCC cross-connection.

The update of the *dccstat* parameter of the two involved DCC channels will be reported via ED-FECOM notification for each port reporting the changed *dccstat* = TRANSPARENT and *scn_dccstat* = TRANSPARENT and *uni_dccstat* = TRANSPARENT and *enni_dccstat* = TRANSPARENT.

Example input/output

The following example shows ENT-CRS-FECOM commands where two transparent DCC cross-connections are established one for *dcctype* = SECTION and one for *dcctype* = LINE:

```
ENT-CRS-FECOM:NE-UNITE:1-1-#-#-1-1,1-1-#-#-2-1,LINE:12345;
    NE-UNITE 00-12-24 12:00:00
 M  123456 COMPLD
;
ENT-CRS-FECOM:NE-UNITE:1-1-#-#-1-1,1-1-#-#-2-1,SECTION:12345;
    NE-UNITE 00-12-24 12:00:00
 M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-CRS-FECOM** command.

Validation rule: Different port AIDs

The AIDs of *port_aid1* and *port_aid2* must specify different ports.

Error code:

`IDNC – Input, Data Not Consistent, port_aid1 and port_aid2 not different`

Validation rule: Maximum of DCC CCs exceeded with CTL/- or CTL/2

Operating on a CTL/- or CTL/2, the maximum number of bidirectional transparent DCC cross-connections is limited to 25.

Error code:

`SSRE – Status, System Resources Exceeded, limit of transparent DCC cross-connections exceeded`

Validation rule: Maximum of DCC CCs exceeded with CTL/3

Operating on a CTL/3 the maximum number of bidirectional transparent DCC cross-connections limited by the following equation: 2 * (nr. of enabled transparent MS/RS-DCC channels) + (nr. of enabled non-transparent MS/RS-DCC channels) + (nr. of internally used DCC channels) ≤ 180.

Error code:

`SSRE – Status, System Resources Exceeded, limit of transparent DCC cross-connections exceeded`

Validation rule: DCC channels enabled

One of the DCC channels is already enabled for DCC overhead processing.

Error code: `SNVS – Status, Not in Valid State, DCC channel already enabled`

Validation rule: DCC protection

One of the DCC channels is occupied the MSP/APS protection of DCC channels.

Error code:

`SNVS – Status, Not in Valid State, DCC channel occupied by MSP/APS DCC protection`

Validation rule: No DCC on transparent ports

The system shall not allow to use DCC on Transparent Ports, i.e. on ports with a rate of STM16T or OC48T.

Error code:

`IDNV – Input, Data Not Valid, DCC on Transparent Ports not allowed`

Validation rule: No DCC on DS3 ports

The system shall not allow to use DCC on DS3 Ports, i.e. on ports with a rate of T3.

Error code: `IDNV – Input, Data Not Valid, DCC on DS3 Ports not allowed`

Validation rule: No DCC on EC1 ports

The system shall not allow to use DCC on EC1 Ports.

Error code: `IDNV – Input, Data Not Valid, DCC on EC1 Ports not allowed`

Related TL1 messages

- "RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection" (p. 3-2329)
- "DLT-CRS-FECOM: Delete transparent DCC cross-connection" (p. 3-156)
- "RTRV-FECOM: Retrieve DCC related parameters" (p. 3-2407)
- "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)

ENT-CRS-FECOM notification

General

Command syntax: **`ENT-CRS-FECOM`**:*port_aid1,port_aid2,dcctype*

Output parameters

Command parameters

1. **`port_aid1`** – DCC port 1
   This parameter indicates one (first) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID
   Range: SPortAID. The keyword **`all`** is allowed not.
2. **`port_aid2`** – DCC port 2
   This parameter indicates one (second) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID.
   Range: SPortAID. The keyword **`all`** is allowed not.
3. **`dcctype`**
   This parameter specifies whether the LINE/MS or SECTION/RS transparent DCC cross-connection parameters of the OP***n*** port is reported by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.

Example output

The following example shows ENT-CRS-FECOM notifications reporting established transparent DCC cross-connections:

```
   UNITE-NE 00-12-24 12:00:00
 A  000 REPT DBCHG
   "TIME=12:00:00,DATE=00-12-24:ENT-CRS-FECOM:1-1-#-#-1-1,1-1-#-2-1,LINE"
;
   UNITE-NE 00-12-24 12:00:00
 A  000 REPT DBCHG
   "TIME=12:00:00,DATE=00-12-24:ENT-CRS-FECOM:1-1-#-#-1-1,1-1-#-2-1,SECTION"
;
```

ENT-EC1 notification

Output format

Command syntax:

ENT-EC1:*aid::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
Type: AID
Range: EC1 port AID

Spec block parameters

1. **crstat** – Constituent Rates Status list
This parameter shows the rates of the port signal.
Type: numeric value set
Range: 11 (1 xSTS1)
2. **eif** – Electrical Interface Type
This parameter indicates the type of input the port has been provisioned to receive.
Type: Alphanumeric value set.
Range: EC1
3. **felpm** – Far-end Line PM Enable
Enable or disable of the performance monitoring of the multiplex section/line.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default)
4. **frcdus** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message "do not use" will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range: ENABLE: set a forced DUS/DNU; DISABLE (default): clear a forced DUS/DNU
5. **fsflpbkstat** – Far Side Facility Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output aid.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback exististing for the aid.
Note: not both loopbacks (lpbk and fsflpbkstat) can be set to YES at the same time.
Type: Alphanumeric value set.
Range: YES, NO (default).

6. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port's bandwidth.
   Type: numeric value set
   Range: 11 (1 xSTS1)
7. **`istd`** – Interface Standard
   Interface standard of the synchronous EC1 interface.
   Type: Alphanumeric value set.
   Range: SONET
8. **`lbo`** – Line Build Out
   This parameter defines the cable length applicable for the EC1 port.
   Type: Alphanumeric value set.
   Range:
   - SHORT (default): 0 … 68.5 m (0 … 225 ft)
   - LONG: 68.5 … 137 m (225 … 450 ft).
9. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output aid.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback exististing for the port.
   Type: Alphanumeric value set.
   Range: YES, NO (default)
10. **`nelpm`** – Near-End Line PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default)
11. **`pmmsfeses`** – Performance Monitoring Line Far-End SES Threshold
    This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
    Type: numeric value set.
    Range: 1 – 6400.
    Default: 51
12. **`pmmsneses`** – Performance Monitoring Line Near-End SES Threshold
    This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
    Type: numeric value set
    Range: 1 – 6400.
    Default: 51
13. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

14. **`pmrsneses`** – Performance Monitoring Section Near-End SES Threshold
    This parameter describes the Section Near-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
    Type: numeric value set.
    Range: 1 … 64000
    Default: 52
15. **`portClass`** – Port Class
    This identifies the port class of the AID.
    Type: Alphanumeric value set.
    Range: TRADITIONAL
16. **`portControl`** – Port Control
    This identifies the administrative mode of the EC1 port. This parameter is not relevant for EC1 ports, because the portClass is always TRADITIONAL. The parameter can only be retrieved but not changed.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
17. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: EC1
18. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when sdsfmode=POISSON.
    Type: numeric value set.
    Range: (as exponents of 10): -5 -6 (default), -7, -8, -9
19. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
20. **`sfthr`** – Signal Failure Threshold
    Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when sdsfmode=POISSON
    Type: numeric value set
    Range: (as exponents of 10): -3 (default), -4, -5

21. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
22. **`sonet`** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports of corresponding line rate (see RTRV-STM4, output parameter sdhpn) to be compatible with current generation applications.
Type: Alphanumeric string
Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
Default: DEFAULT.
23. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
24. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc).
Type: Numeric string.
Range: 1 for specific repeating byte (SONET)
25. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted.
Type: Hexadecimal digits
Range: 1 byte string, One byte is expressed as two hexadecimal digits (example value: 6D).
Default: 01Hex
26. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (*strcout*).
Type: Numeric string.
Range: 1 for specific repeating byte (SONET)

27. **trbmd** – Tributary mode
This parameter indicates the tributary mode of the port. In the case of EC1 port, the only possible value is FIXED.
Type: Alphanumeric value set.
Range: FIXED

28. **unequiposig** – Unequipped Output Signal List
Tributary Unequipped Output Signal Rate List.
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection.
Type: numeric value set
Range: 11 (1 x STS1)

Example Output

```
   UNITE-NE 04-12-23 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=04-12-23:ENT-EC1:1-1-#-#-21-1:: eif=EC1,ptype=EC1,sfthr=-
      3,sdthr=-6, pmode=MON,..... pmmsneses=51,pmrsneses=52″
;
```

ENT-ENNIMANROUTE: Enter manual route to the E-NNI node

The **ENT-ENNIMANROUTE** command is used on ASON domain border NEs, that means, NEs that have ENNI interfaces to other ASON domains, to enter (add) manual routes to other ASON domains into the forwarding table of the E-NNI node.

There are two types of manual routes:

- manual routes to neighboring signaling controllers
- manual routes to neighboring routing controllers

The command modifier is used to distinguish these two types of manual routes.

Note that this command cannot be used to enter manual routes to neighboring signaling controllers that are connected via point-to-point interfaces, as these manual routes are implicitely added upon port provisioning.

Refer to the Example Input/Output section for provisioning examples.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-ENNIMANROUTE**-modifier*:tid::ctag:::[spec_block];*

Input parameters

Command parameters

1. **`modifier`**
   The parameter modifier indicates whether the manual route is used to reach a signaling controller (SC) or a routing controller (RC) subnet.
   Type: Alphanumeric value set.
   Range: SC or RC
2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`neighborSIGPCIPAddress`** – Neighbor Signaling Controller IP Address
   This parameter is used to indicate the IP address of a neighboring signaling controller that is connected via a LAN interface.
   The term "neighbor" refers to logical neighborship. The neighboring SC is not necessarily a physical neighbor.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
2. **`neighborSIGPCID`** – Neighbor Signaling Controller Identifier
   This parameter is used to indicate the identifier of a neighboring signaling controller that is connected via a LAN interface.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.

3. **`neighborRCIPAddress`** – Neighbor Route Controller IP Address
   This parameter is used to indicate the IP address of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
4. **`neighborRCIPSubnetMask`** – Neighbor Route Controller IP Subnet Mask
   This parameter is used to indicate the subnet mask of the IP subnet of a neighboring route controller.
   The IP subnet that can be reached via the manual route is described by the tuple neighborRCIPAddress + neighborRCIPSubnetMask.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
5. **`lan_nh_ifipad`** – Next Hop Interface IP Address for LAN Interfaces
   This parameter is used to indicate the interface IP address of the next hop of a manual route using a LAN. This parameter is insignificant for point-to-point interfaces.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the interface refered by *if_id* is a LAN.
6. **`if_id`** – interface id
   This parameter is used to indicate the interface associated with the manual route.
   Type: AID
   Range: SPortAID or LanPort AID (the keyword “all” is not used)
7. **`dcctype`** – DCC type
   This parameter is used to further specify the DCC type, in case the *if_id* refers to a DCC channel.
   Type: Alphanumeric value set.
   Range: LINE or SECTION
   Optional parameter: This parameter must be present if the interface *if_id* is a DCC. It must not be present, if the interface *if_id* is no DCC.

Output format

If the **`ENT-ENNIMANROUTE`** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Detailed behavior description

The **ENT-ENNIMANROUTE** command shall add a manual route entry to the E-NNI node. The update of the manual routes are notified with ENT-ENNIMANROUTE Notifications.

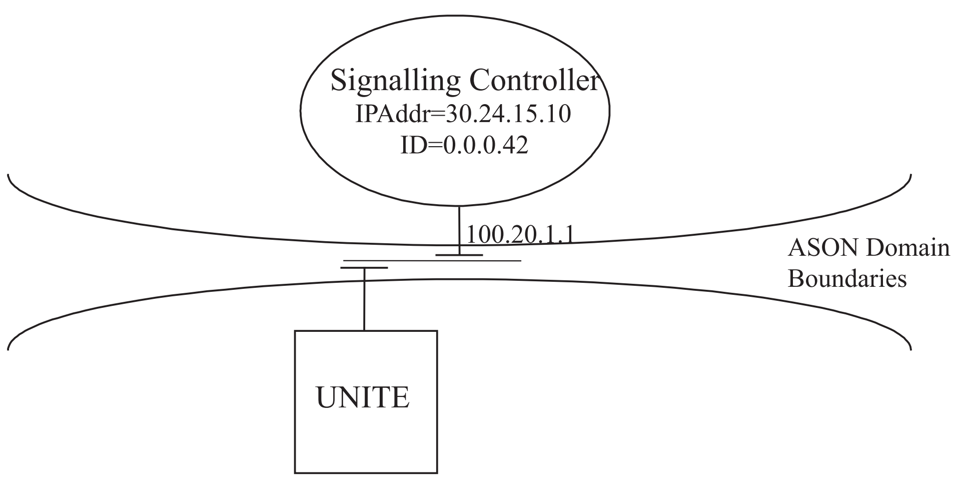
Example input/output

```
ENT-ENNIMANROUTE-RC:LT-UNITE::123456:::neighborRCIPAddress=95.22.10.1,
      neighborRCIPSubnetMask=255.255.255.0, lan_nh_ifipad=100.20.1.1, if_id=1-1-
      lan2;

   LT-UNITE 06-10-09 08:00:00
M  123456 COMPLD
;
```

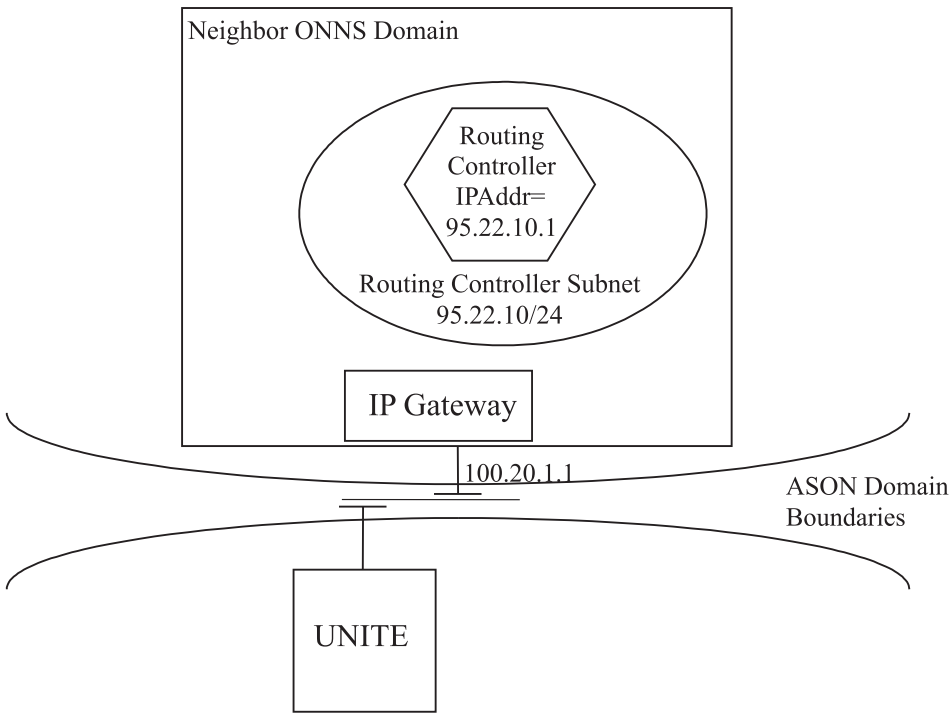
Signaling Controller Provisioning Example

```
ENT-ENNIMANROUTE-SC:LT-
UNITE::123456:::neighborSIGPCIPAddress=30.24.15.10,
neighborSIGPCID=0.0.0.42, lan_nh_ifipad=100.20.1.1,
if_id=1-1-lan2;
```



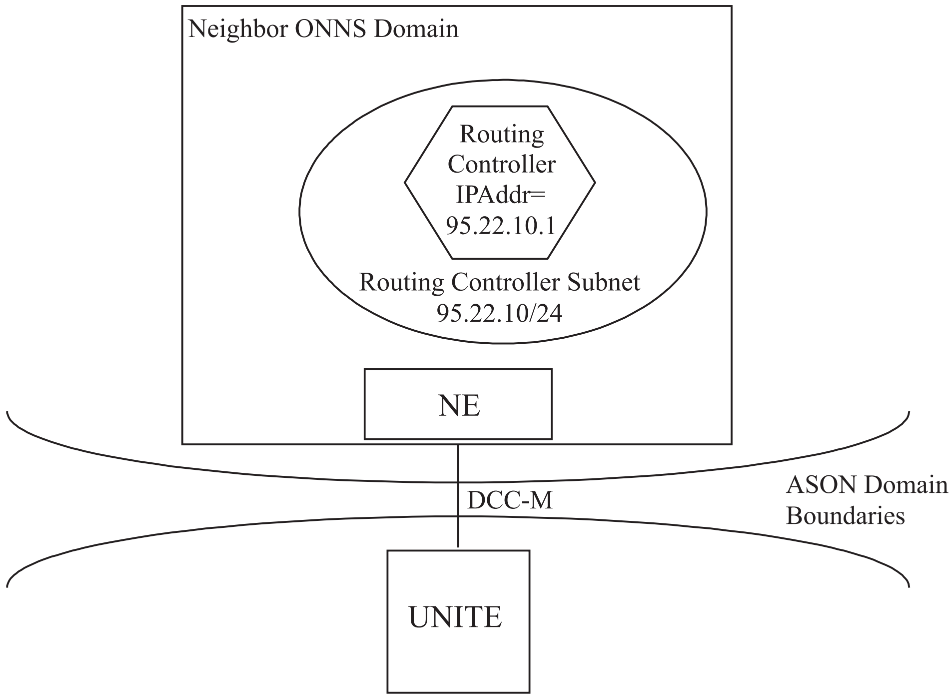
Routing Controller Provisioning Example P2MP

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0,
lan_nh_ifipad=100.20.1.1, if_id=1-1-lan2;
```



Routing Controller Provisioning Example P2P

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0, if_id=1-1-#-#-5-3,
dcctype=LINE;
```



Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ENT-ENNIMANROUTE** command.

Validation rule: Parameter Combinations

The command is rejected if the combination of modifier and parameters is not one of the following:

- modifier==SC and if is a LAN and list of parameters is: *neighborSIGPCIPAddress*, *neighborSIGPCID*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a LAN and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a point-to-point interface and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, *dcctype*

Error code:

`IDNV – Input, Data Not Valid, Combination of Modifier/Parameters not allowed`

Validation rule: Not allowed to use LAN1 for E-NNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for ENNI`

Validation rule: LAN port not used for E-NNI

If the LAN port specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LAN port not in use for ENNI`

Validation rule: DCC port not used for E-NNI

If the DCC specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, DCC not in use for ENNI`

Validation rule: No duplicate Manual Route entries

The command is denied, if a duplicate manual route entry is entered.

A manual route is considered to be duplicate if one of the following conditions is fulfilled:

- an entry with identical *neighborSIGPCIPAddress*, *neighborSIGPCID*, and *if_id* is already provisioned
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, and *if_id* is already provisioned, where *if_id* refers to a LAN interface
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, and *dcctype* is already provisioned, where *if_id* refers to a DCC

Error code:

`IDNV – Input, Data Not Valid, Manual route entry already exists`

Note that the *lan_nh_ifipad* is by intention not considered, as it is not allowed to have more than one gateway to one destination on one and the same LAN.

Validation rule: Check if limit of 20 Manual Routes per LAN port exceeded

The addition of a manual route entry for a LAN port is denied, if already 20 manual route entries exist for the LAN port.

Error code:

`SSRE – Status, System Resources Exceeded, limit of manual route entries exceeded`

Validation rule: Check if limit of 64 Manual Routes per ENNI Node exceeded

The addition of a manual route entry is denied, if already 64 manual route entries exist for the ENNI node.

Error code:

`SSRE – Status, System Resources Exceeded, ENNI limit of manual route entries exceeded`

This requirement is independent from the type of interface (LAN or point-to-point).

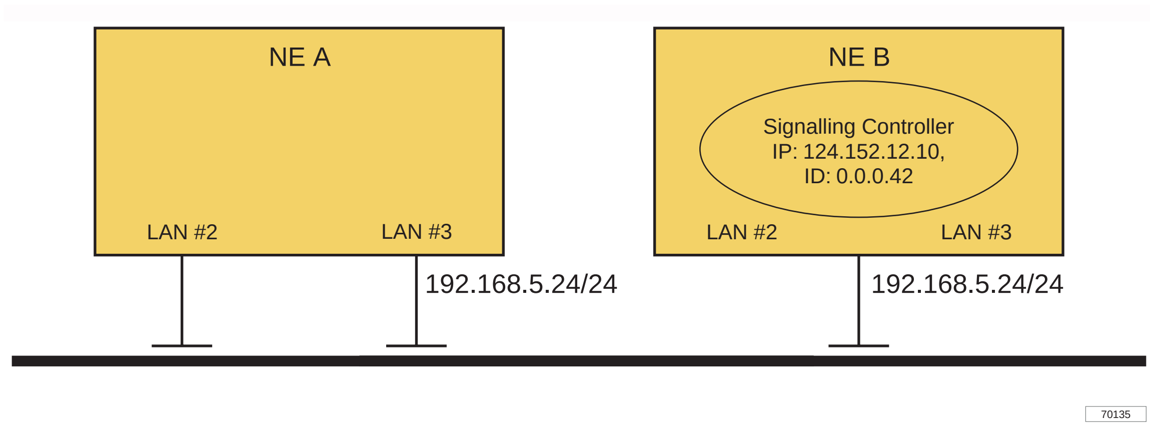
Validation rule: No more than one LAN connected to same next hop

The command is rejected if the provided *lan_nh_ifipad* is already provisioned for any other LAN interface.

Error code:

`IDNV – Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`

This requirement prevents a configuration as shown below, which is desirable, but currently not supported:



Related TL1 messages

- "RTRV-ENNIMANROUTE: Retrieve all manual routes of the E-NNI node" (p. 3-2348)
- "DLT-ENNIMANROUTE: Delete manual route from the E-NNI node" (p. 3-164)

ENT-ENNIMANROUTE notification

Output format

The **ENT-ENNIMANROUTE** notification has the following specific syntax:

ENT-ENNIMANROUTE-modifier*:::spec_block*

Output parameters

Command parameters

1. **modifier**
   The **ENT-ENNIMANROUTE** command is used on ASON domain border NEs, i.e. NEs that have ENNI interfaces to other ASON domains, to enter (add) manual routes to other ASON domains into the forwarding table of the E-NNI node.
   There are two types of manual routes:
   - manual routes to neighboring signaling controllers
   - manual routes to neighboring routing controllers

   The command modifier is used to distinguish these two types of manual routes.
   Note that this command cannot be used to enter manual routes to neighboring signaling controllers that are connected via point-to-point interfaces, as these manual routes are implicitely added upon port provisioning.
   Refer to the Example Input/Output section for provisioning examples.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-ENNIMANROUTE**-modifier*:tid::ctag:::[spec_block];*

Input parameters

Command parameters

1. **`modifier`**
   The parameter modifier indicates whether the manual route is used to reach a signaling controller (SC) or a routing controller (RC) subnet.
   Type: Alphanumeric value set.
   Range: SC or RC
2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`neighborSIGPCIPAddress`** – Neighbor Signaling Controller IP Address
   This parameter is used to indicate the IP address of a neighboring signaling controller that is connected via a LAN interface.
   The term "neighbor" refers to logical neighborship. The neighboring SC is not necessarily a physical neighbor.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
2. **`neighborSIGPCID`** – Neighbor Signaling Controller Identifier
   This parameter is used to indicate the identifier of a neighboring signaling controller that is connected via a LAN interface.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.

3. **`neighborRCIPAddress`** – Neighbor Route Controller IP Address
   This parameter is used to indicate the IP address of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
4. **`neighborRCIPSubnetMask`** – Neighbor Route Controller IP Subnet Mask
   This parameter is used to indicate the subnet mask of the IP subnet of a neighboring route controller.
   The IP subnet that can be reached via the manual route is described by the tuple neighborRCIPAddress + neighborRCIPSubnetMask.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
5. **`lan_nh_ifipad`** – Next Hop Interface IP Address for LAN Interfaces
   This parameter is used to indicate the interface IP address of the next hop of a manual route using a LAN. This parameter is insignificant for point-to-point interfaces.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the interface refered by *if_id* is a LAN.
6. **`if_id`** – interface id
   This parameter is used to indicate the interface associated with the manual route.
   Type: AID
   Range: SPortAID or LanPort AID (the keyword “all” is not used)
7. **`dcctype`** – DCC type
   This parameter is used to further specify the DCC type, in case the *if_id* refers to a DCC channel.
   Type: Alphanumeric value set.
   Range: LINE or SECTION
   Optional parameter: This parameter must be present if the interface *if_id* is a DCC. It must not be present, if the interface *if_id* is no DCC.

Output format

If the **`ENT-ENNIMANROUTE`** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Detailed behavior description

The **ENT-ENNIMANROUTE** command shall add a manual route entry to the E-NNI node. The update of the manual routes are notified with ENT-ENNIMANROUTE Notifications.

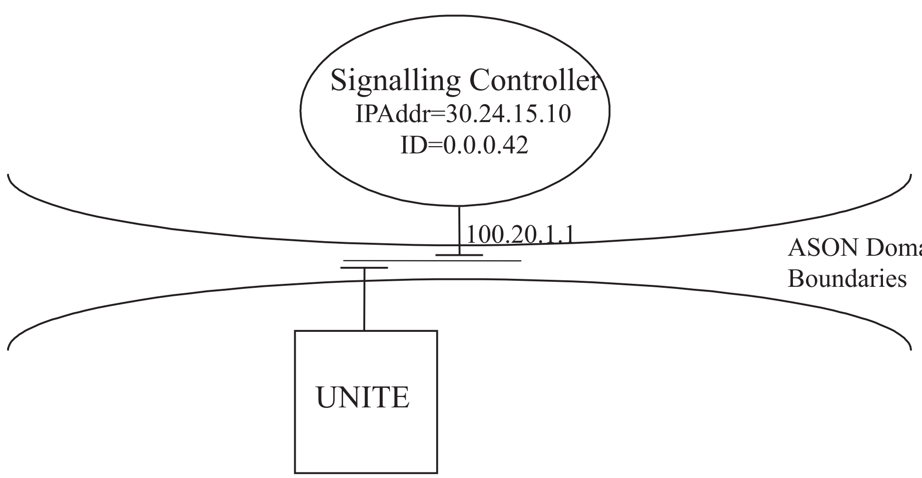
Example input/output

```
ENT-ENNIMANROUTE-RC:LT-UNITE::123456:::neighborRCIPAddress=95.22.10.1,
      neighborRCIPSubnetMask=255.255.255.0, lan_nh_ifipad=100.20.1.1, if_id=1-1-
      lan2;

   LT-UNITE 06-10-09 08:00:00
M  123456 COMPLD
;
```

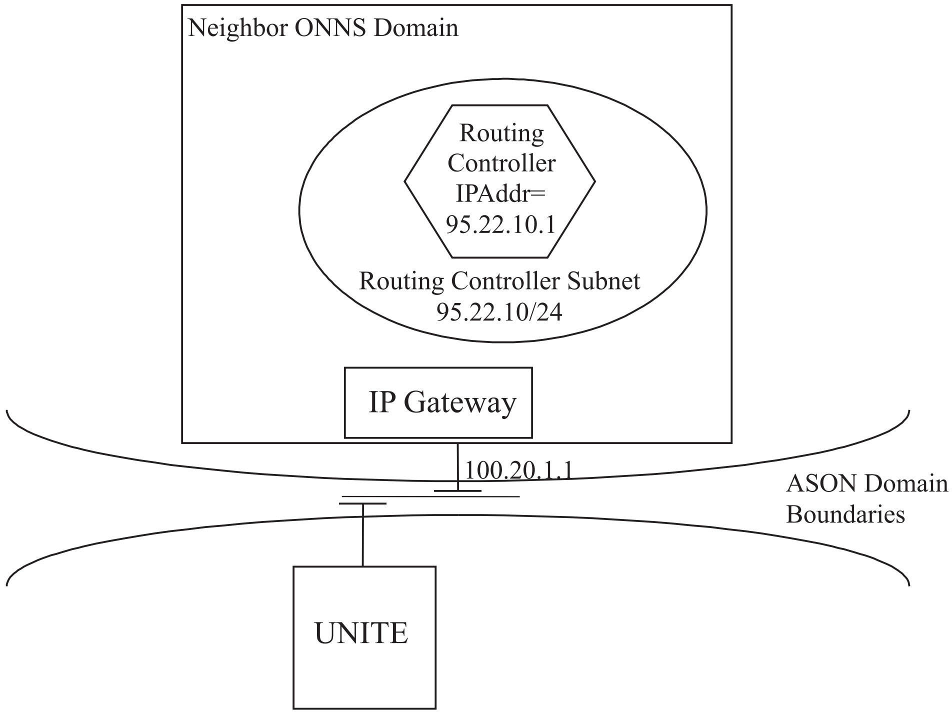
Signaling Controller Provisioning Example

```
ENT-ENNIMANROUTE-SC:LT-
UNITE::123456:::neighborSIGPCIPAddress=30.24.15.10,
neighborSIGPCID=0.0.0.42, lan_nh_ifipad=100.20.1.1,
if_id=1-1-lan2;
```



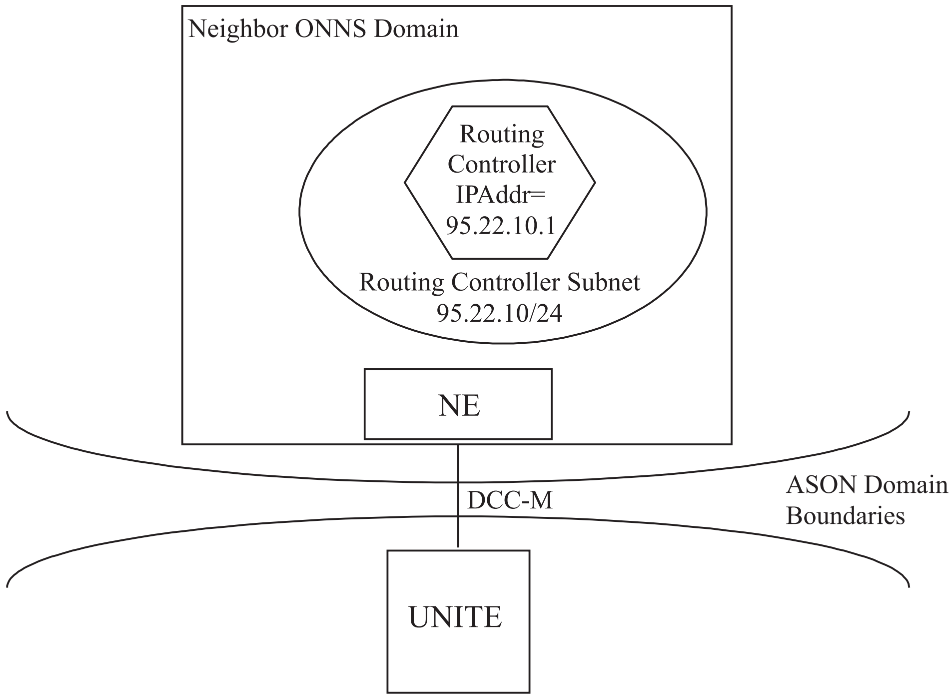
Routing Controller Provisioning Example P2MP

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0,
lan_nh_ifipad=100.20.1.1, if_id=1-1-lan2;
```



Routing Controller Provisioning Example P2P

```
ENT-ENNIMANROUTE-RC:LT-
UNITE::123456:::neighborRCIPAddress=95.22.10.1,
neighborRCIPSubnetMask=255.255.255.0, if_id=1-1-#-#-5-3,
dcctype=LINE;
```



Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ENT-ENNIMANROUTE** command.

Validation rule: Parameter Combinations

The command is rejected if the combination of modifier and parameters is not one of the following:

- modifier==SC and if is a LAN and list of parameters is: *neighborSIGPCIPAddress*, *neighborSIGPCID*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a LAN and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *lan_nh_ifipad*, *if_id*
- modifier==RC and interface is a point-to-point interface and list of parameters is: *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, *dcctype*

Error code:

`IDNV – Input, Data Not Valid, Combination of Modifier/Parameters not allowed`

Validation rule: Not allowed to use LAN1 for E-NNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for ENNI`

Validation rule: LAN port not used for E-NNI

If the LAN port specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LAN port not in use for ENNI`

Validation rule: DCC port not used for E-NNI

If the DCC specified in the AID is not in use for E-NNI, i.e. is not assigned to the enninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, DCC not in use for ENNI`

Validation rule: No duplicate Manual Route entries

The command is denied, if a duplicate manual route entry is entered.

A manual route is considered to be duplicate if one of the following conditions is fulfilled:

- an entry with identical *neighborSIGPCIPAddress*, *neighborSIGPCID*, and *if_id* is already provisioned
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, and *if_id* is already provisioned, where *if_id* refers to a LAN interface
- an entry with identical *neighborRCIPAddress*, *neighborRCIPSubnetMask*, *if_id*, and *dcctype* is already provisioned, where *if_id* refers to a DCC

Error code:

`IDNV – Input, Data Not Valid, Manual route entry already exists`

Note that the *lan_nh_ifipad* is by intention not considered, as it is not allowed to have more than one gateway to one destination on one and the same LAN.

Validation rule: Check if limit of 20 Manual Routes per LAN port exceeded

The addition of a manual route entry for a LAN port is denied, if already 20 manual route entries exist for the LAN port.

Error code:

`SSRE – Status, System Resources Exceeded, limit of manual route entries exceeded`

Validation rule: Check if limit of 64 Manual Routes per ENNI Node exceeded

The addition of a manual route entry is denied, if already 64 manual route entries exist for the ENNI node.

Error code:

`SSRE – Status, System Resources Exceeded, ENNI limit of manual route entries exceeded`

This requirement is independent from the type of interface (LAN or point-to-point).

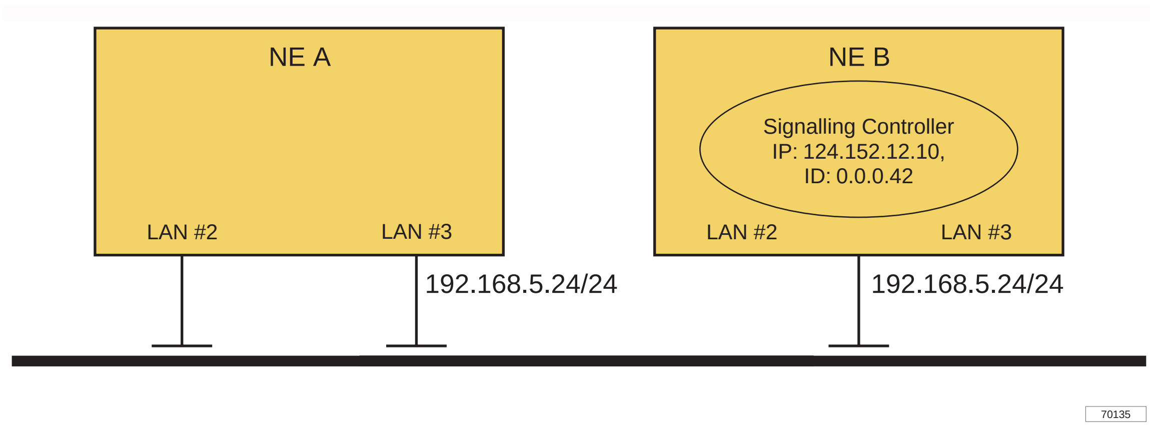
Validation rule: No more than one LAN connected to same next hop

The command is rejected if the provided *lan_nh_ifipad* is already provisioned for any other LAN interface.

Error code:

`IDNV – Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`

This requirement prevents a configuration as shown below, which is desirable, but currently not supported:



Related TL1 messages

- “RTRV-MANROUTE-LAN: Retrieve manual route at LAN port” (p. 3-2516)
- “DLT-MANROUTE-LAN: Delete manual route at LAN port” (p. 3-203)

ENT-EPORT notification

Output format

The **ENT-EPORT** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax:

ENT-EPORT-modifier*:aid::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the Ethernet port to be operated on.
   Type: AID
   Range: EPortAID as defined in AID structure. The usage of the following values are excluded
   - *all* in the BayAID definition
   - *all* in the Simple_ShelfAID definition
   - *usall* in the Simple_UnivSlotAID definition
   - *all* in the Simple_UnivSlotAID definition
   - *all* in the Simple_EPortAID definition.
2. **modifier**
   This parameter specifies the type of the Ethernet port at the LAN side of the unit.
   Type: Alphanumeric value set.
   Range:
   - 1GE representing a 1 GE port on a GE1 pack
   - 1GEPL representing a 1 GE port on a GE10PL1 pack
   - 10GEPL representing a 10 GE ports on a GE10PL1 pack.

Spec block parameters

1. **`autoneg`** – Auto Negotiation Mode
   This parameter controls auto-negotiation process to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE or 10GEPL. If the parameter *autoneg* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default); DISABLE.
2. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer LAN port or a network LAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER (default), for a customer LAN port; NETWORK, for a network LAN port.
3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (i.e. from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - 1, 2, 3, … 4092, 4093
   - UNAVAILABLE (default).

   To express the state UNAVAILABLE system internally the invalid VLAN ID 4094 is used.

5. **`droppause`** – Drop Incoming Pause Frames
   This parameter controls whether incoming pause frames are dropped (value = ENABLE) or not (value = DISABLE). This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter droppause is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
   Default: DEFAULT.
6. **`enetpn`** – Ethernet Alarm Severity Assignment Profile
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
7. **`enettca`** – Ethernet TCA Profile
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for Ethernet ports.
   Type: Alphanumeric string
   Range: 1 … 24 characters length
   Default: DEFAULT0
8. **`epmode`** – Ethernet Port Monitoring Mode
   This parameter specifies the monitoring mode of the Ethernet port. Alarms for this port will be reported if in state MON otherwise not. The configuration of epmode will change from AUTO to MON if a valid signal is applied to that port.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default), for automatic
   - MON, for monitored
   - NMON, for not monitored.
9. **`eppm`** – Ethernet Port Performance Monitoring
   This parameter specifies whether the performance monitoring is enabled or disabled for this port.
   Type: Alphanumeric value set.
   Range:
   - DISABLE (default)
   - ENABLE
10. **`fcmd`** – Flow Control Mode
    This parameter specifies the configuration for the pause operation of this port. If the input parameter modifier = 1GE, then the parameter fcmd is supported only in case that the bridge operates in backward-compatible mode (vlantagmd = BCMD).
    Type: Alphanumeric value set.

Range:

- AUTO, for auto-negotiate
- DISABLE, for pause mode bidirectionally disabled
- ASYMTOPARTNER, for pause mode having receive disabled, transmit enabled
- ASYMTOLOCAL, for pause mode having transmit disabled, receive enabled
- SYM, for pause mode bidirectionally enabled.

Restrictions: If the parameter *fcmd* is not supported, then the parameter is omitted.
The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GE.
The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = ENABLE. The value AUTO is notsupported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = DISABLE.
The value AUTO is not supported, if the input parameter *modifier* = 10GEPL.
Possible combinations of the parameters *modifier*, *autoneg*, *vlantagmd* (parameter of the referring circuit pack entity, see "RTRV-EQPT (CP): Retrieve equipment (circuit pack)" (p. 3-2375) for definition) and *fcmd* are as follows:

Input parameter *modifier*	**In-/output parameter *autoneg***	**In-/output parameter *vlantagmd***	**In-/output parameter *fcmd***
1GE	not supported	BCMD	AUTO, DISABLE
		other than BCMD	not supported
1GEPL	ENABLE	not supported	AUTO, DISABLE
	DISABLE		DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM
10GEPL	not supported	not supported	DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM

Default:

- AUTO (for *modifier* 1GE or 1GEPL)
- SYM (for *modifier* 10GEPL).

11. **`fsflpbkstat`** – Far Side Facility Loopback Status
This parameter reports the configuration of the far-side facility loopback of this port. A far-side loopback means that the signal on the outgoing port is looped to the incoming port (Tx -> Rx). For GE1 there is only this type of facility loopback.
Type: Alphanumeric value set.
Range:
    - YES, for enabled loopback
    - NO (default), for disabled loopback.

12. **`rcf_htmr`** – Remote Client Fail Hold Off Timer
    The parameter specifies the hold-off time for applying the LPT consequent action to disable the laser at the LAN port. The hold-off timer is only applicable if there is a strict one-to-one relationship between the LAN port in question and its related WAN port.
    This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_htmr* is not supported, then the parameter is omitted.
    Type: numeric string [ms].
    Range: 0 … 2000 ms in 100 ms steps
    Default: 0.
13. **`rcf_linkdown`** – Link Pass Through Mode
    This parameter controls the consequent action (to switch-off the laser upon receipt of a client signal fail [CSF] indication) to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter rcf_linkdown is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: ENABLE (default), DISABLE.
14. **`rptmode`** – Repeater Mode
    This parameter sets the bridge device for this LAN port into the repeater mode (promiscuous mode), valid for a point to point connection between one LAN and one WAN port. Operating in repeater mode imposes some restrictions. See detailed behavior section.
    If the input parameter *modifier* = 1GE, then the parameter rptmode is meaningful only if in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter cnprole is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.

    Range:

    - ENABLE
    - DISABLE (default).
15. **`servcond`** – Service Condition
    This parameter specifies the service condition for this port. Some configuration possibilities like loopbacks may be dependent on the state of *servcond*.
    Type: Alphanumeric value set.

    Range:

    - INSERVICE (default)
    - OUTOFSERVICE.

16. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. If the input parameter modifier = 1GE, then this parameter is meaningful only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter stdis is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - DISABLE, STP is disabled, i.e. the port is not used in the spanning tree protocol
    - ENABLE (default), STP is enabled, i.e. the port is used in the spanning tree protocol.
17. **`stpcost`** – Spanning Tree Path Cost
    The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
    If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
    If the parameter *stpcost* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 to 200000000.
    Default: 20000.
    The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.
18. **`stpp`** – Spanning Tree Port Priority
    This parameter specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.

The stepping as recommended by the standard (steps of 16 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 240″ is not done by the NE.
Default: 128.

19. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding. Two thresholding methods are applicable: TR-only. (also known as the implicit reset method), TR-RTR. (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR (default).

20. **`vrtswid`** – Virtual Switch Identifier
This parameter identifiers the instance of the virtual switch.
    - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
    - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

    Type: Integer.
    Range: 1 – Currently this is the only allowed value.
    Default: 1.

21. **`vvidls`** – Valid VLAN Identifier List (static)
This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using ’&’ and ’&&’ . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (cnprole = NETWORK).
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter vvidls is not supported, then the parameter is omitted.
Type: Alphanumeric string and the character ’&’.

Range:

- Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
- In case the list has no members (empty list) -1.

The length of the list is limited only by the usual system limitations like e.g. TL1 line length. The number of entries in this list is limited to 64 by some NE-internal restrictions. Ranging and grouping by using the operators ’&’ and ’&&’ has no impact on the number of entries, it only reduces the length of the list being transported to/from CIT/EMS.
Default: -1.
Example 1:
vvidls=-1. The list has no members (empty list).
Example 2:
vvidls=3&5&40&&45. This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.
Example 3:
vvidls=1&&4093 This list has the members 1,2, 3, …, 4092, 4093.

Example Output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:ENT-EPORT-1GE:1-1-#-#-2-3::\
    cnprole=CUSTOMER,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,\
    enettca=DEFAULT,epmode=AUTO,eppm=DISABLE,fcmd=AUTO,\
    fsflpbkstat=NO,rptmode=DISABLE,rstpmd=ENABLE,\
    servcond=INSERVICE,stdis=DISABLE,stpcost=20000,stpp=128,\
    tcamd=TR,vrtswid=1,vvidls=-1”
;
```

ENT-EQPT (CP) notification

Output format

Syntax: **`ENT-EQPT`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The AID of an equipment.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for circuit pack in any slot
   - LoxcAID::=UnivSlotAID-loxc for LOXC pack in universal slots.

   The keyword **`usall`** is not allowed for Simple_UnivSlotAID.
   The keyword **`xcall`** is not allowed for Simple_XCSlotAID.
   The keyword **`ctlall`** is not allowed for Simple_CTLSlotAID.
   The keyword **`all`** is not allowed for Simple_SlotAID.

Spec block parameters

1. **`dcn_ipad`** – DCN IP address for CLI/SNMP
   DCN IP address for CLI/SNMP management of the card.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255.
   The values 0.0.0.0, 1.1.1.1, 127.0.0.1, and 224.0.0.0 to 255.255.255.255 are excluded.
   Default: Empty string.
2. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the circuit pack. This parameter is reported for a CpAID , PowerAID, EciAID, TimingInterfaceAID and LoxcAID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
3. **`eqptprotpn`** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT for the equipment.
   Option: This parameter is only reported for a CpAID, LoxcAID or PowerAID..
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
   Default: DEFAULT

4. **`es64ucap`** – ES64U capacity
   Switch capacity of the ES64U pack.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0 – ES64U in suspended state, initial value after ES64U CP creation.
   - 20 – ES64U active or member of equipment protection group, capacity 20 Gbit/s.

   Default: 0.
5. **`opstate`** – Operational State
   Each circuit pack has a retrievable attribute indicating whether circuit pack is operational for the system internal communication.
   Type: Alphanumeric value set.
   Range:
   - OS_ENABLED
   - OS_DISABLED (default)
   - OS_SYNCHING.
6. **`loswcap`** – LOXC switch capacity
   Switch capacity of the LOXC pack. This parameter is relevant only for circuit packs of type LOXC, thus the parameter shall be omitted for other packs.
   Type: value set.
   Range, if *nvmname* = LOXC
   - 0: LOXC in suspended state, initial value after LOXC CP creation
   - 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s.

   Range, if *nvmname* = LOXC40G2S or LOXC40G3S:
   - 0: LOXC in suspended state, initial value after LOXC CP creation
   - 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s
   - 40: LOXC active or member of equipment protection group, switch capacity 40 Gbit/s

   Default: 0.
7. **`lostd`** – LO tributary standard
   LO tributary standard supported by the LOXC pack. This parameter is relevant only for circuit packs of type LOXC (i.e. nvmname = LOXC).
   Type: Value set.

Range:

- SDH: LOXC supports SDH LO tributaries (VC3, VC12)
- SONET: LOXC supports SONET LO tributaries (VT1.5)
- UNDEFINED (default): no LO tributaries (loswcap = 0)

LO VC3 tributaries provided with lostd=SDH support LO cross-connections and AU3/AU4 conversion cross-connections.

8. **`tmuxcap`** – TransMUX capacity
   Switch capacity of the TransMUX pack.
   This parameter is relevant only for circuit packs of type TransMUX (that means, *nvmname* = TransMUX), thus the parameter shall be omitted for other packs.
   Type: Value set.
   Range:
   - 0: TransMUX in suspended state, initial value after TransMUX CP creation.
   - 2G5: TransMUX active or member of equipment protection group, capacity 2.5 Gbit/s

   Default: 0
9. **`istd`** – Interface Standard of the Circuit Pack
   Interface standard supported by the circuit pack. This parameter is supported on GE10PL1 and ES64U circuit packs. On the circuit packs, the parameter *istd* defines the interface standard of the virtual concatenation groups.
   Option: If the parameter *istd* is not supported, then the parameter is omitted.
   Type: Value set.
   Range:
   - for GE10PL1: SDH or SONET
   - for ES64U: SONET or UNDEFINED (ES64U not used, *es64ucap* = 0)

   Default:
   - for GE10PL1: The default is derived from the system parameter *istddflt*.
   - for ES64U: UNDEFINED.

Example output

```
   UNITE-NE 00-11-14 08:00:30
A  001 REPT DBCHG
   ”TIME=08-00-12,DATE=00-11-14:ENT-EQPT:1-1-#-#-1-
      cp:::eqptpn=DEFAULT,eqptprotpn=DEFAULT,opstate=OS_ENABLE”;
```

ENT-EQPT (slot): Enter equipment (slot)

The **ENT-EQPT** command pre-provisions a slot for a dedicated circuit pack.

The CTL can not be pre-provisioned. It is always auto-provisioned.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-EQPT***:tid:aid:ctag:::[spec_block];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a slot.
   Type: AID
   Range: SlotAID
   The keyword **usall** is not allowed for Simple_UnivSlotAID.
   The keyword **xcall** is not allowed for Simple_XCSlotAID.
   The keyword **ctlall** is not allowed for Simple_CTLSlotAID.
   The keyword **all** is not allowed for Simple_SlotAID, Simple_UnivSlotAID, Simple_XCSlotAID, Simple_CTLSlotAID.
   The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`nvmname`** – Circuit Pack Name
   This parameter specifies the name of the provisioned circuit pack of the slot. If no circuit pack is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.
2. **`nvmqual`** – Circuit Pack Qualifier
   This parameter specifies the qualifier of the provisioned circuit pack of the slot. If no circuit pack is provisioned the value UNDEF is reported.
   Type: value set.
   Range: 7 characters.
   The allowed values are:
   - UNDEF (default)
   - EXTD
   - all values as defined in SRD_ConfigurationManagement

   Note: The TL1-representation of the functional qualifier follows the following rules:
   - alphanumeric characters or the special character '-' only
   - special characters like '/' and '.' to be omitted.

Output format

If the **`ENT-EQPT`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command preprovisions a slot for a dedicated circuit pack. The following attributes of the slot entity is updated due to that operation:

1. *nvmname* as specified in the command input
2. *nvmqual* as specified in the command input.

The command is only allowed if the slot is not provisioned (*nvmname/nvmqual* = UNDEFINED/UNDEF).

The update is reported by means of an **`ED-EQPT`** notification for that slot.

Behavior of *actlname* during provisioning

When a slot is provisioned for a dedicated circuit pack, then the actlname of the provisioned slot is updated and only contain the functional names of circuit packs which are allowed for reprovisioning. The parameter actlname shall at least contain the functional name of the provisioned circuit pack, independent if reprovisioning is supported for the provisioned circuit pack.

Behavior of preprovisioning of multislot circuit packs

The attribute *actlname* of the neighbor slots which are also covered by a multislot circuit pack (e.g. OP40, LOXC40G2S, LOXC40G3S) is modified and set to empty (i.e. the output contains one row with *actlname*=″″, see also Example Input/output section of RTRV-EQPT(slot)).

The attributes *nvmname*, *nvmqual* of the neighbor slots which are also covered by a multislot circuit pack (e.g. OP40, LOXC40G3S) is modified and set to *nvmname*, EXTD.

The update is reported by means of an **ED-EQPT** notification for those slots.

Behavior of preprovisioning on multislot circuit packs

The attribute *actlname* of the neighbor slot which can not be used anymore by a multislot circuit pack (the physical slot of e.g. an OP40 or LOXC40G2S, LOXC40G3S) is modified and all multislot CPs are removed.

The update is reported by means of an **ED-EQPT** notification for this slot.

Changes for neighbor slots when provisioning double-density circuit packs

When provisioning the first LOXC pack the attribute *actlname* of slots which are allowed to carry LOXC packs is modified according to defined rules. The update is reported by means of an **ED-EQPT** notification for the affected slots.

Changes for neighbor slots when provisioning an LOXC40G3S pack

If the system parameter *swmaxcap* is set to LXC320 or LXC160, the provisioning of a 3-slot LOXC40G3S pack in slot n shall clear the acceptable CP name list of the neighbor slot n-3. The update is reported by means of an **ED-EQPT** notification for slot n-3.

Changes for slots allowing to carry any type of LOXC pack when provisioning an LOXC pack

When provisioning the first LOXC pack, the attribute *actlname* of slots which are allowed to carry LOXC packs is modified. The update is reported by means of an **ED-EQPT** notification for the affected slots.

Changes for slots allowing to carry a TransMUX pack when provisioning a TransMUX pack

When provisioning the TransMUX pack, the attribute *actlname* of slots which are allowed to carry TransMUX packs is modified. The update is reported by means of an **`ED-EQPT`** notification for the affected slots.

Changes for slots allowing to carry a ES64U pack when provisioning a ES64U pack

When provisioning the ES64U pack, the attribute *actlname* of slots which are allowed to carry ES64U packs is modified. The update is reported by means of an **`ED-EQPT`** notification for the affected slots.

Update affected entities

The command creates in addition a set of entities of different types depending on the value of *nvmname* of command input as follows.

1. One instance of type circuit pack is created. This is reported by means of an ENT-EQPT notification for circuit pack.
2. Some instances of port STM-16 are created if *nvmname* of command equals *OP2G5* and attribute *istddflt* of the node equals *SDH*. This is reported by means of ENT-STM16 notification for port. The number of port instances to be created depends on the capabilities of the actual circuit pack.
3. Some instances of port OC-48 are created if *nvmname* of command equals *OP2G5* and attribute *istddflt* of the node equals *SONET*. This is reported by means of ENT-OC48 notification for port. The number of port instances to be created depends on the capabilities of the actual circuit pack.
4. One instance of port STM-64 is created if *nvmname* of command equals *OP10* and attribute *istddflt* of the node equals *SDH*. This is reported by means of ENT-STM64 notification for port.
5. One instance of port OC-192 is created if *nvmname* of command equals *OP10* and attribute *istddflt* of the node equals *SONET*. This is reported by means of ENT-OC192 notification for port.
6. One instance of port STM-256 is created if *nvmname* of command equals *OP40* and attribute *istddflt* of the node equals *SDH*. This is reported by means of ENT-STM256 notification for port.
7. One instance of port OC-768 is created if *nvmname* of command equals *OP40* and attribute *istddflt* of the node equals *SONET*. This is reported by means of ENT-OC768 notification for port.
8. 48, 192, 768 VC3 tributaries are created for each to the previously created SDH port according to the port rate. This is reported by means of ENT-VC3 notification for tributary.
9. 48, 192, 768 STS1 tributaries are created for each to the previously created SONET port according to the port rate. This is reported by means of ENT-STS1 notification for tributary.

10. Two dcc channels entities are created (one per type section/dccr and one type line/dccm) for each previously created SDH/SONET port. This is reported by means of ENT-FECOM notification for dcc channel.
11. One instance of 1+1 equipment protection group of type 'EQPTSWFBR' is created if nvmname of command equals *XC320* and another *XC320* circuit pack exists already. This is reported by means of ENT-PROTNGRP notification for 1+1 equipment protection group. The initial protection state is reported by means of a protection switch notification for the 1+1 equipment protection group.

Update affected entities – GE1

The command creates in addition a set of entities of different types depending on the value of *nvmname* of command input as follows.

1. One instance of type circuit pack is created. This is reported by means of an **`ENT-EQPT`** notification for circuit pack.
2. Four instances of entity EPORT are created if *nvmname* of command equals *GE1*. This are reported by means of ENT-EPORT notification.
3. Four instances of entity VCG are created if *nvmname* of command equals *GE1*. This are reported by means of ENT-VCG notification.
4. 7 VC-4 tributaries are created for each to the previously created VCG if attribute *istddflt* of the node equals 'SDH'. This is reported by means of ENT-VCGTRIB notification.
5. 21 STS-1 tributaries are created for each to the previously created VCG if attribute *istddflt* of the node equals 'SONET'. This is reported by means of ENT-VCGTRIB notification.

Update affected entities – GE10PL1

The command shall create in addition a set of entities of different types depending on the value of nvmname of command input as follows:

1. One instance of type circuit pack is created. This is reported by means of an **`ENT-EQPT`** notification for circuit pack.
2. Nine instances of type socket is created (one for each socket contained on the circuit pack). This are reported by means of ENT-EQPT notifications for sockets.

Update affected entities - Port Entities of OP155

In addition to the rules listed in ID 17111 the command shall create the following entities if nvmname equals OP155:

- Some instances of port STM-1 are created if the attribute *istddflt* of the node equals SDH. This is reported by means of ENT-STM1 notification for port. The number of port instances to be created depends on the capabilities of the actual circuit pack.
- Some instances of port OC-3 are created if the attribute *istddflt* of the node equals SONET. This is reported by means of ENT-OC3 notification for port. The number of port instances to be created depends on the capabilities of the actual circuit pack.

- 3 VC3 tributaries are created for each of the previously created SDH ports. This is reported by means of ENT-VC3 notifications for tributary.
- 3 STS1 tributaries are created for each of the previously created SONET port. This is reported by means of ENT-STS1 notifications for tributary.

Update affected entities - Port Entities of EP155

In addition to the rules listed in ID 17111 the command shall create the following entities if nvmname equals EP155:

- 8 STM-1E port instances of the actual circuit pack is created. This are reported by means of ENT-STM1E notification for port.
- 1 VC4 tributary is created for each of the previously created SDH ports. This are reported by means of ENT-VC3 notifications for tributary.

Update affected entities - OPLB

The command shall create following entities:

- One instance of type circuit pack is created. This is reported by means of an **`ENT-EQPT`** notification for circuit pack.
- One instance of type socket is created for each socket contained on the circuit pack one. This are reported by means of ENT-EQPT notifications for sockets.

Update affected entities - OP2G5D

The command shall create following entities:

- One instance of type circuit pack is created. This is reported by means of an **`ENT-EQPT`** notification for circuit pack.
- One instance of type socket is created for each socket contained on the circuit pack one. This are reported by means of ENT-EQPT notifications for sockets.

Update affected entities - OPT2G5

The command creates the following entities:

- One instance of type circuit pack is created. This is reported by means of an **`ENT-EQPT`** notification for circuit pack.
- Three instances of type socket is created (one for each socket contained on the circuit pack). This are reported by means of ENT-EQPT notifications for sockets.
- Three instances of entity VCG-TRANS is created. This are reported by means of ENT-VCG-TRANS notification.
- 17 VC-4 tributaries are created for the previously created VCG-TRANS if the attribute istddflt of the node equals ’SDH’. This are reported by means of ENT-VCGTRIB notification.
- 17 STS-3c tributaries are created for the previously created VCG-TRANS if the attribute istddflt of the node equals ’SONET’. This are reported by means of ENT-VCGTRIB notification.

Update affected entities - OP10D

The command creates the following entities:

- One instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.
- One instance of type socket is created for each socket contained on the circuit pack. This are reported by means of ENT-EQPT notifications for sockets.

Update affected entities - LOXC

The command creates the following entities:

- One instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.

Update affected entities - EP51

The command creates the following entities:

- One instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.
- 36 instances of port T3 is created. This are reported by means of **ENT-T3** notification for port.
- 1 STS1 path termination is created for each of the previously created T3 ports. This are reported by means of ENT-PTF-STS notifications.

Update affected entities - TransMUX

The command creates the following entities:

- The instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.

Update affected entities - ES64U

The command creates the following entities:

- The instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.

Update affected entities - OP10XTTC and OP10XTTL

The command creates the following entities:

- One instance of type circuit pack is created. This is reported by means of an **ENT-EQPT** notification for circuit pack.
- One OTN port of type OTU2 is created. This are reported by means of an ENT-OTU2 notification for OTN port.
- One instance of port STM-64 is created if the parameter istddflt of the system equals ’SDH’. The parameter otnclient of the STM-64 port is set to YES. This are reported by means of an ENT-STM64 notification for port.

- One instance of port OC-192 is created if the parameter istddflt of the system equals 'SONET'. The parameter otnclient of the OC-192 port is set to YES. This are reported by means of an ENT-OC192 notification for port.
- 192 VC3 tributaries are created for previously created SDH port. This are reported by means of an ENT-VC3 group notification for tributary.
- 192 STS1 tributaries are created for the previously created SONET port. This are reported by means of an ENT-STS1 group notification for tributary.
- Two DCC channels entities are created (one per type section/dccr and one type line/dccm) for the previously created SDH/SONET port. This are reported by means of ENT-FECOM notifications for dcc channels.

Group notification for STS-1/AU-3 tributaries

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to:

1&&-<N> with N is maximal legal STS-1/AU-3 number.

AID range for group notification: See STS-1/AU-3 Tributaries AID w.r.t. range for legal STS-1/AU-3 number.

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

Group notification for VCG tributaries

For efficiency reasons the notifications for VCG tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the used mapping of the associated VCG port, i.e. the Simple_VCGNumAID field in the VCGNumAID is set to:

- 1&&-<N> for sequential range
- 1&-4 .. &-<N> for non-sequential range.

with N is maximal legal VCG tributary number.

AID range for group notification: See VCG tributaries AID w.r.t. range for legal VCG number and sequential/non-sequential aid range.

Example: The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like: 1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping) 1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping).

Example input/output

```
ENT-EQPT:UNITE-NE:1-1-#-#-1:123456:::nvmname=OP10,nvmqual=1.5LR1;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
```

;

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-EQPT** command.

Validation rule: Slot must not be provisioned

It should be checked whether the slot is still auto or preprovisioned. Only a unprovisioned slot can be preprovisioned for a dedicated circuit pack. To preprovision a slot the *nvmname/nvmqual* is UNDEFINED/UNDEF (=unprovisioned).

Error code: `IDNV – Input, Data Not Valid, Slot still provisioned`

Validation rule: Parameter *nvmname* must be specified

The system shall check that the parameter *nvmname* is specified in the command.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Check whether CP is acceptable

Check of the circuit pack type which is allowed for the slot.

Consistency is checked at two levels:

- between *actlname*/*actlqual* with *nvmname*/*nvmqual* at system level
- between actlname with nvmname at slot level.

Error code:
`EATN – Status, incorrect equipage, unrecognized CP for this slot`

Validation rule: Check whether system is maintenance condition

Preprovisioning is not allowed, when the system in maintenance condition (see “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400)).

Error code:
`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Reject Request for CTL slots

If *ctlw* or *ctlp* is specified as Simple_CTLSlotAID the command is rejected.

Error code: `IENE – Input, Entity Not Exists`

Related TL1 messages

- “ED-EQPT (slot): Edit equipment (slot)” (p. 3-370)
- “RTRV-EQPT (slot): Retrieve equipment (slot)” (p. 3-2386)
- “DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
- “RTRV-ACTLCP: Retrieve acceptable circuit pack list” (p. 3-2222).

References

GR-199-CORE

ENT-EQPT (socket): Enter equipment (socket)

The **ENT-EQPT** command preprovisions a socket for a dedicated module.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-EQPT**:*tid*:*aid*:ctag:::*[spec_block];*

Input parameters

Command parameters

1. **tid** – Target Identifier
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
The AID of a socket.
Type: AID.
Range: SocketAID. The keywords *all* / *usall* / *scall* shall are not allowed on any level.
3. **ctag** – Correlation Tag
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **nvmname** – Provisioned Module Name
This parameter specifies the name of the provisioned module of the socket. If no module is provisioned the default value is reported.
Type: value set.
Range: 12 character.
Default: UNDEFINED.
2. **nvmqual** – Provisioned Module Qualifier
This parameter specifies the qualifier of the provisioned module of the slot. If no module is provisioned the default value is reported..
Type: value set.
Range: 7 characters.
Default: UNDEF.

Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

- alphanumeric characters or the special character ’-’ only
- special characters like ’/’ and ’.’ to be omitted.

Output format

If the **ENT-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command shall preprovision a socket for a dedicated module.

The following attributes of the socket entity is updated due to that operation:

- *nvmname* as specified in the command input
- nvmqual as specified in the command input.

The command is only allowed if the socket is unprovisioned (nvmname/nvmqual = UNDEFINED/UNDEF).

The update is reported by means of an **ED-EQPT** notification for that socket.

Update acceptable modules list

If the command leads to a change of the Acceptable Modules List (list of actlname/actlqual), then the change is reported by means of an **ED-EQPT** notification. Since the creation of the module is not reported no ENT-EQPT notification for a module entity will be reported.

Update affected entities

The command creates in addition a set of entities of different types depending on the value of *nvmname* of command input as follows.

- One instances of port STM-16 is created if *nvmname* of command equals ’OM2G5’ and attribute *istddflt* of the node equals ’SDH’. This are reported by means of ENT-STM16 notification for port.
- One instances of port OC-48 is created if *nvmname* of command equals ’OM2G5’ and attribute *istddflt* of the node equals ’SONET’. This are reported by means of ENT-OC48 notification for port.

- One instance of port STM-64 is created if *nvmname* of command equals ’OM10’ and attribute *istddflt* of the node equals ’SDH’. This are reported by means of ENT-STM64 notification for port.
- One instance of port OC-192 is created if *nvmname* of command equals ’OM10’ and attribute *istddflt* of the node equals ’SONET’. This are reported by means of ENT-OC192 notification for port.
- One instance of port STM-4 is created if *nvmname* of command equals ’OM622’ and attribute *istddflt* of the node equals ’SDH’. This are reported by means of ENT-STM4 notification for port.
- One instance of port OC-12 is created if *nvmname* of command equals ’OM622’ and attribute *istddflt* of the node equals ’SONET’. This are reported by means of ENT-OC12 notification for port.
- One instance of port STM-1 is created if *nvmname* of command equals ’OM155’ and attribute *istddflt* of the node equals ’SDH’. This are reported by means of ENT-STM1 notification for port.
- One instance of port OC-3 is created if *nvmname* of command equals ’OM155’ and attribute *istddflt* of the node equals ’SONET’. This are reported by means of ENT-OC3 notification for port.
- 3, 12, 48, 192 VC3 tributaries are created for each to the previously created SDH port according to the port rate. This are reported by means of ENT-VC3 notification for tributary.
- 3, 12, 48, 192 STS1 tributaries are created for each to the previously created SONET port according to the port rate. This are reported by means of ENT-STS1 notification for tributary.
- Two dcc channels entities are created (one per type section/dccr and one type line/dccm) for each previously created SDH/SONET port. This are reported by means of ENT-FECOM notification for dcc channel.

Update affected entities - OPT2G5

If the socket resides on a OPT2G5 circuit pack, then the following entities are created:

- One instance of port STM-16T is created if the attribute istddflt of the node equals ’SDH’. This are reported by means of ENT-STM16T notification for port.
- One instance of port OC-48 is created if the attribute istddflt of the node equals ’SONET’. This are reported by means of ENT-OC48T notification for port.

Update affected entities - GE10PL1

If the socket resides on a GE10PL1 circuit pack, then the following entities are created:

- One instance of 1 Gb Ethernet port is created if nvmname of the command equals ’OMGE1’. This are reported by means of ENT-EPORT notification for Ethernet port.
- One instance of 10 Gb Ethernet port is created if nvmname of the command equals ’OMX10’. This are reported by means of ENT-EPORT notification for Ethernet port.
- One instance of entity VCG is created for the previously created Ethernet port. If another VCG already exists on the GE10PL1 pack, then the interface standard for the VCG to be created shll be derived from the existing VCG. Otherwise, if no VCG exixts on the GE10PL1 pack, the interface standard for the VCG to be created shll be derived from the system’s interface standard default. The creation of the VCG is reported by means of ENT-VCG notification.
- 21 VC-3 tributaries are created for the previously created VCG if the interface standard of the VCG equals ’SDH’ and the VCG is related to a 1 GbE port. This are reported by means of ENT-VCGTRIB notification.
- 21 STS-1 tributaries are created for the previously created VCG if the interface standard of the VCG equals ’SONET’ and the VCG is related to a 1 GbE port. This are reported by means of ENT-VCGTRIB notification.
- 192 VC-3 tributaries are created for the previously created VCG if the interface standard of the VCG equals ’SDH’ and the VCG is related to a 10 GbE port. This are reported by means of ENT-VCGTRIB notification.
- 192 STS-1 tributaries are created for the previously created VCG if the interface standard of the VCG equals ’SONET’ and the VCG is related to a 10 GbE port. This are reported by means of ENT-VCGTRIB notification.

Group notification for STS-1/AU-3 tributaries

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to:

- 1&&-<N>

with N is maximal legal STS-1/AU-3 number.

Example:

The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

Example input/output

This example shows an ENT-EQPT command for pre provisioning of a socket:

```
ENT-EQPT:UNITE-NE:1-1-#-#-5-sc2:123456:::nvmname=OM2G5,nvmqual=13LR1;

   UNITE-NE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-EQPT** command.

Validation rule: Socket must not be provisioned

It should be check whether the socket is still auto or preprovisioned. Only a unprovisioned slot can be preprovisioned for a dedicated module. To preprovision a socket the nvmname/nvmqual is UNDEFINED/UNDEF (=unprovisioned).

Error code: `IDNV – Input, Data Not Valid, Socket still provisioned`

Validation rule: Parameter *nvmname* must be specified

The system shall check that the parameter *nvmname* is specified in the command.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Check whether system is maintenance condition

Pre-provisioning is not allowed, when the system in maintenance condition. (See “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400).)

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Equipage Status and Inventory Data of Module must be accessible

If the equipage status or inventory data of a module is not accessible (e.g. due to the parent pack being in recovery) pre-provisioning of the socket is not allowed.

Error code: `SNVS – Status, Not in Valid State, unknown module equipage`

Validation rule: Module must be plugged out

It is checked whether no module is plugged in. This implies that also no unexpected module is plugged in.

Error code:

`SNVS – Status, Not in Valid State, the module is not plugged out`

Validation rule: Module must be part of acceptable module list

It is checked whether the module to be pre-provisioned is part of the acceptable module list (parameters *actlname*, *actlqual* of the addressed socket).

Error code:
`ENRE – Equipage, Not Recognized Equipage, unrecognized module for this socket`

Related TL1 messages

- “DLT-EQPT (socket): Delete equipment (socket)” (p. 3-188)
- “ED-EQPT (socket): Edit equipment” (p. 3-379)
- “RTRV-EQPT (socket): Retrieve equipment (socket)” (p. 3-2391)

References

GR-199-CORE

ENT-EQPT (socket) notification

Output format

Syntax: **ENT-EQPT**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **all** is not allowed on bay/shelf level.

Spec block parameters

1. **eqptpn** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT on socket level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
   Default: DEFAULT.
2. **nvmname** – Provisioned Module Name
   This parameter specifies the name of the provisioned module of the socket. If no module is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.
3. **nvmqual** – Provisioned Module Qualifier
   This parameter specifies the qualifier of the provisioned module of the slot. If no module is provisioned the default value is reported..
   Type: value set.
   Range: 7 characters.
   Default: UNDEF.

   Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

   - alphanumeric characters or the special character ’-’ only
   - special characters like ’/’ and ’.’ to be omitted.

Example output

The following examples shows the case after pre-provisioning of slot 5 socket 2:

```
A 001 REPT DBCHG
  ”TIME=08-00-12,DATE=00-11-14:ENT-EQPT:1-1-#-#-5-
     sc2::nvmname=OM2G5,nvmqual=15LR1,eqptpn=PROFILE1”
```

```
;
```

ENT-FECOM: Enter DCC channel related parameters

The **ENT-FECOM** command allows to enable or disable a DCC channel for a specific OSI node, the SCN node, the UNI node or the E-NNI node. It can also be used to modify the type of LAPD service of an existing DCC channel, which is used for MCN.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-FECOM**:*tid*::*ctag*::*port*[,*dcctype*]:[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **port**
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID.
   Range: SPortAID; the keyword **all** is not allowed.
4. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC of the *port* is affected by the command.
   Type: Alphanumeric value set.
   Range:
   - LINE
   - SECTION.

   Default: SECTION if *dccstat* is set to ENABLE. No default (i.e. parameter is mandatory) if *scn_dccstat* or *uni_dccstat* is ENABLE.

Spec block parameters

1. **`dccstat`**
   This parameter indicates whether communication over the DCC channel is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat* or *enni_dccstat* must be present in the command.
2. **`lapdmode`**
   This parameter indicates the type of LAPD service. Modes are AITS and UITS. Starting with R2.0 this parameter is also used to determine the service for the LinkID protocol.
   If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* is set to ENABLE, or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set..
   Range:
   - UITS (default)
   - AITS.
3. **`lapdrl`**
   This parameter indicates the LAPD role. This parameter is also used to determine the service for the LinkID protocol.
   If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* is set to ENABLE, or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set.
   Range:
   - USER-SIDE (default)
   - NETWORK-SIDE.
4. **`osinode`**
   Assignment of a DCC channel to an MCN node in the NE. This parameter must be supplied when dcc_stat = ENABLED and not the default osinode1 is used, in all other cases it is omitted.
   An NE can have up to 8 instances of the OSI stack. Node-1 is default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00). Nodes 2-8 can be created and configured, see "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951).
   Type: AID.
   Range: osinode{1-8}
   Default: osinode1.

5. **`scn_dccstat`**
   This parameter indicates whether the DCC channel is used for communication over the SCN.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat* or *enni_dccstat* must be present in the command.
6. **`uni_dccstat`**
   This parameter indicates whether the DCC channel is used for UNI communication.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat* or *enni_dccstat* must be present in the command.
7. **`enni_dccstat`**
   This parameter indicates whether the DCC channel is used for E-NNI communication.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat* or *enni_dccstat* must be present in the command.

Output format

If the **`ENT-FECOM`** command is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **`ENT-FECOM`** command can assign a DCC channel to a specified node and enable or disable it. Furthermore the type of LAPD service can be set respectively modified for UITS (default) or AITS.

The update is reported by means of **`ED-FECOM`** notification.

It seems to be odd that an ENT command leads to an ED notification. The rationale behind that is as follows: **`ENT-FECOM`** has edit semantics, but it wasn’t going to be changed into an ED command due to backward compatibility to CG. The notification behavior was, however, clean-up. Additional **DLT-FECOM**, **ENT-FECOM** notification are defined to notify the deletion/creation of DCC channels upon CP deletion/creation or MSP DCC slaving enabled/disabled.

Note: If *scn_node* is set to ENABLED, the channel is implicitely assigned to the SCN node.

Note: If *uni_node* is set to ENABLED, the channel is implicitely assigned to the UNI node.

Note: If *enni_node* is set to ENABLED, the channel is implicitly assigned to the E-NNI node.

The re-assignment of a dedicated DCC channel from an osinode to the scnnode or vice versa is not supported within one command execution.

A re-assignment from *osinode* to *scnnode* means: Set from *dccstat* = ENABLE to *scn_dccstat*={ENABLE|DISABLE}.

A re-assignment from *scnode* to *osinode* means: Set from *scn_dccstat* = ENABLE to *dccstat*={ENABLE|DISABLE}.

The re-assignment can be achieved in three steps:

- disable the channel for the old node
- change associated port type
- enable the channel for a new node.

The re-assignment of a dedicated DCC channel from an osinode or the scnnode to the uninode or vice versa is not supported within one command execution.

A re-assignment from osinode or scnnode to uninode means: *uni_dccstat* is specified in the command while either *dccstat* or *scn_dccstat* is set to ENABLE.

A re-assignment from uninode to osinode or scnnode means: *dccstat* or *scn_dccstat* is specified in the command while *uni_dccstat* is set to ENABLE.

The re-assignment can be achieved in two steps:

- disable the channel for the old node
- enable the channel for a new node.

The re-assignment of a dedicated DCC channel from an osinode or the scnnode or the uninode to the enninode or vice versa is not supported within one command execution.

A re-assignment from osinode or scnnode or uninode to enninode means: *enni_dccstat* is specified in the command while either *dccstat* or *scn_dccstat* or *uni_dccstat* is set to ENABLE.

A re-assignment from enninode to uninode or osinode or scnnode means: *dccstat* or *scn_dccstat* or *uni_dccstat* is specified in the command while *enni_dccstat* is set to ENABLE.

The re-assignment can be achieved in three steps:

- disable the channel for the old node
- change associated port type
- enable the channel for a new node.

When *dccstat* is set from DISABLED to ENABLED, the system shall apply the default values for all spec block parameters except *scn_dccstat* and *uni_dccstat* which are not specified in the command.

In contrast to the general pre-provisioning concept, it is not possible to change any parameter when not one of *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* is set to ENABLED.

It is possible to specify additional parameters in the command, which is used to set *dccstat* to DISABLE. The parameter values are changed as specified, but when one of *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* is set to ENABLE, all spec block parameters are set to default values again.

Example input/output

The following example shows the result of ENT-FECOM:

```
ENT-FECOM:NE-UNITE::123456::1-1-#-#-5-
      1,line:dccstat=enable,lapdmode=AITS,osinode=osinode5;
   LT-UNITE 00-12-24 12:00:00
M  123456 COMPLD
;
```

The following example shows the result of ENT-FECOM command which enables a DCC channel for the SCN:

```
ENT-FECOM:NE-UNITE::123456::1-1-#-#-5-1,line:scn_dccstat=ENABLE;
   LT-UNITE 00-12-24 12:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-FECOM** command.

Validation rule: Check for that either *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* is present

The command is rejected when none of the parameters *dccstat*, *scn_dcnstat*, *uni_dccstat*, or *enni_dccstat* is present.

Error code: `IDNV – Input, Data Not Valid`

Validation rule: Check for enabled DCC channels

The command is rejected when any other parameter than *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* is changed and one of the parameters *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* is set to DISABLE with the command. The validation is not applied, if *dccstat* is set to DISABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: Single occurence of *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* parameter

The command is rejected when more than one parameter of *dccstat*, *scn_dccstat*, *uni_dccsstat*, or *enni_dccsstat* is present in the command.

Error code:

`IDNC – Input, Data Not Consistent, only one of dccstat, scn_dccstat, uni_dccstat, or enni_dccstat allowed`

Validation rule: Implicit status change of *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat*

Only the actually enabled channel can be set to disabled, i.e. the command is rejected:

- when *dccstat* = ENABLE it is not allowed to set *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* to DISABLE,
- when *scn_dccstat* = ENABLE it is not allowed to set *dccstat*, *uni_dccstat*, or *enni_dccstat* to DISABLE,
- when *uni_dccstat* = ENABLE it is not allowed to set *dccstat*, *scn_dccstat*, or *enni_dccstat* to DISABLE,
- when *enni_dccstat* = ENABLE it is not allowed to set *dccstat*, *scn_dccstat*, or *uni_dccstat* to DISABLE.

Error code:

`SNVS – Status, Not in Valid State, implicit channel re-assignment`

Validation rule: Check for parameter osinode

The command is rejected when *dccstat* is set to ENABLED and *osinode* is not in the created state. This means a value for parameter *osinode* other than osinode1 is used in the command, which was not used for creation of a node via ENT-ULSDCC-L3 before.

Error code: `ENEQ – Equipage, Not EQuipped osinode`

Validation rule: DCC Protection

A port cannot be specified when it is protecting part of a 1+1 MSP group and DCC protection switching is enabled.

Error code: `SNVS – Not in Valid State, part of 1+1 MSP protection`

When DCC protection switching is enabled, the RS/MS DCC channel objects of the protection port are deleted and not available.

Validation rule: Check for DCC Protection

Disabling of RS-DCC/MS-DCC is rejected when RS-DCC/MS-DCC protection dcc is enabled.

Error code: `SNVS – Not in Valid State, DCC protection available`

Validation rule: DCC for 2 fiber BLSR/MS-SPRING

DCC-disabling on a port of a 2-fiber BLSR/MS-SPRING protection group is denied if no other DCC channel is available on that port although automatic ring discovery is enabled.

Error code: `SNVS – Not in Valid State, DCC needed for BLSR/MS-SPRING`

Validation rule: Worker port for 4 fiber BLSR/MS-SPRING

DCC-disabling on the port of the worker line of a symmetric span of a 4-fiber BLSR/MS-SPRING protection group is denied if no other DCC channel is available on that port although automatic ring discovery is enabled.

Error code: `SNVS – Not in Valid State, DCC needed for BLSR/MS-SPRING`

Validation rule: Protection port for 4 fiber BLSR/MS-SPRING

DCC-disabling on the port of the protection line of an asymmetric span of a 4-fiber BLSR/MS-SPRING protection group is denied if no other DCC channel is available on that port although automatic ring discovery is enabled.

Error code: `SNVS – Not in Valid State, DCC needed for BLSR/MS-SPRING`

Validation rule: Check for service condition

Enable DCC termination is not possible if port is in out-of-service state.

Error code: `SDNC – Status, Data Not Consistent, port is out-of-service`

Validation rule: Transparent DCC cross-connection

The DCC channel is not ENABLED or DISABLED if it is used in a transparent DCC cross-connection. *dccstat*, *scn_dccstat*, *uni_dccstat* and *enni_dccstat* cannot be changed until the existing transparent DCC cross-connection is deleted.

Error code:

```
SNVS – Status, Not in Valid State, DCC channel used in transparent DCC
cross-connection
```

Validation rule: Independent use of DCC channel for MCN, SCN, UNI, or E-NNI

The command is rejected when another channel than the actual enabled channel is set to enabled, i.e.:

- when *dccstat* = ENABLE it is not allowed to set *scn_dccstat*, *uni_dccstat* or *enni_dccstat* to ENABLE,
- when *scn_dccstat* = ENABLE it is not allowed to set *dccstat*, *uni_dccstat* or *enni_dccstat* to ENABLE,
- when *uni_dccstat* = ENABLE it is not allowed to set *dccstat*, *scn_dccstat* or *enni_dccstat* to ENABLE,
- when *enni_dccstat* = ENABLE it is not allowed to set *dccstat*, *scn_dccstat* or *uni_dccstat* to ENABLE.

Error code:

```
SNVS – Status, Not in Valid State, DCC channel can not be assigned to
more than one of MCN, SCN, UNI and E-NNI at the same time
```

Validation rule: Maximum number of channels assigned to MCN on CTL/-

Operating on a CTL/- the system can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for MCN (i.e. dccstat=ENABLE) is not exceeded.

Error code:

```
SSRE – Status, System Resources Exceeded, overall limit of DCC channels
exceeded
```

Validation rule: Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL/2

Operating on a CTL/2, the system can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for MCN, SCN, UNI, or E-NNI (i.e. *dccstat* = ENABLE or *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE or *enni_dccstat* = ENABLE) is not exceeded.

Error code:

```
SSRE – Status, System Resources Exceeded, overall limit of DCC channels
exceeded
```

Validation rule: Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL4

Operating on a CTL/4T or CTL/4S, the system can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for MCN, SCN, UNI, or E-NNI (i.e. *dccstat* = ENABLE or *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE or *enni_dccstat* = ENABLE) is not exceeded.

Error code:

`SSRE – Status, System Resources Exceeded, overall limit of DCC channels exceeded`

Validation rule: Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL/3

Operating on a CTL/3 the system can support up to 180 DCC interfaces according to the following rule:

2 * (number of enabled transparent MS/RS-DCC channels) + (number of enabled non-transparent MS/RS-DCC channels) + (number of internally used MS/RS-DCC channels) ≤ 180. It is checked that this rule is not violated.

Error code:

`SSRE – Status, System Resources Exceeded, overall limit of DCC channels exceeded`

Internally used DCC channels for ES64U pack:

Each unprotected ES64U pack provisioned with a capacity of 20G and each equipment protected pair of ES64U packs in the system uses 1 DCC-M.

Validation rule: No DCC on transparent ports

The system shall not allow to use DCC on Transparent Ports, i.e. on ports with a rate of STM16T or OC48T.

Error code:

`IDNV – Input, Data Not Valid, DCC on Transparent Ports not allowed`

Validation rule: No DCC on DS3 ports

The system shall not allow to use DCC on DS3 Ports, i.e. on ports with a rate of T3.

Error code: I`DNV – Input, Data Not Valid, DCC on DS3 Ports not allowed`

Validation rule: No DCC on EC1 ports

The system shall not allow to use DCC on EC1 Ports.

Error code: I`DNV – Input, Data Not Valid, DCC on EC1 Ports not allowed`

Validation rule (SCN specific): Check for provisioned SCN IP address

The command is rejected when scn_dccstat=ENABLED and no IP address has been configured for the SCN via ENT-ULS (ip_ad_astn).

Error code:

`SNVS – Status, Not in Valid State, no SCN IP address provisioned`

Validation rule (SCN specific): Check combination of *osinode* parameter with *scn_dccstat*

The command is rejected when *scn_dccstat* = ENABLED and a value for parameter *osinode* is supplied.

Error code:

`SDNC – Status, Data Not Consistent, invalid combination of osinode parameter with dccstat, scn_dccstat`

Validation rule (SCN specific): Check whether ONNS is enabled

The command is rejected when *scn_dccstat* = ENABLE in the command, while ONNS has not been enabled (system parameter *xcsetup* not set to SWISS).

Error code: `SNVS – Status, Not in Valid State, SwiSS not enabled`

Validation rule (SCN specific): RS-DCC not allowed for SCN

The command is rejected if *scn_dccstat* = ENABLED and *dcctype* = SECTION..

Error code:

`IDNV – Input, Data Not Valid, DCC type SECTION not allowed for SCN`

Validation rule (SCN specific): Check consistency of input parameters with *portClass* I

The command is rejected, if *scn_dccstat* = ENABLED and the port is not of *portClass* INNI.

Error code: `IDNV – Input, Data Not Valid, portClass must be INNI`

Validation rule (SCN specific): Invalid input parameters

The command is rejected, if scn_dccstat=ENABLED and a value for *lapdmode* or a value for *lapdrl* is given.

Error code:

`IDNV – Input, Data Not Valid, provisioning of lapdmode and lapdrl not possible`

Validation rule (SCN specific): Prevent that both DCCs of a port are assigned to the SCN

The command is rejected, if it is tried to set *scn_dccstat* = ENABLED for a specific port and *dcctype* and for the same port *scn_dccstat* is already ENABLED for the other *dcctype*.

Error code:

`IDNV – Input, Data Not Valid, not possible to assign both channels to SCN`

Validation rule (SCN specific): Prevent that both DCCs of a port are assigned to the MCN, when *portClass* = INNI

The command is rejected, if *portClass* = INNI for port and it is tried to set *dccstat* = ENABLED for a specific port and *dcctype* and for the same port *dccstat* is already ENABLED for the other *dcctype*.

Error code:

`IDNV – Input, Data Not Valid, not possible for NNI port to assign both channels to MCN`

Validation rule (SCN specific): Maximum number of channels assigned to SCN

The scnnode can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for scnnode (i.e. *scn_dccstat* = ENABLED) is not exceeded.

Error code:

`SSRE – Status, System Resources Exceeded, allowed limit of SCN DCC channels exceeded`

Validation rule (UNI/E-NNI specific): Check combination of osinode parameter with *uni_dccstat / enni_dccstat*

The command is rejected when any of the following expressions is true:

- *uni_dccstat* = ENABLED and (*enni_dccstat* = ENABLED or a value for parameter osinode is supplied),
- *enni_dccstat* = ENABLED and (*uni_dccstat* = ENABLED or a value for parameter osinode is supplied),
- *scn_dccstat* = ENABLED and (*uni_dccstat* = ENABLED or *enni_dccstat* = ENABLED or a value for parameter osinode is supplied),
- *dccstat* = ENABLED and (*scn_dccstat* = ENABLED or *uni_dccstat* = ENABLED or *enni_dccstat* = ENABLED).

Error code:

`SDNC – Status, Data Not Consistent, invalid combination of osinode parameter with dccstat, scn_dccstat, uni_dccstat, enni_dccstat`

Validation rule (UNI/E-NNI specific): Check consistency of input parameters with *portClass*

The command is rejected, if *uni_dccstat* = ENABLED and the port is not of *portClass* UNI.

Error code: `IDNV – Input, Data Not Valid, portClass must be UNI`

Validation rule (UNI/E-NNI specific): Check consistency of input parameters with *portClass*

The command is rejected, if *enni_dccstat* = ENABLED and the port is not of *portClass* UNI.

Error code: `IDNV – Input, Data Not Valid, portClass must be ENNI`

Validation rule (UNI/E-NNI specific): E-NNI Node must be created

The command is rejected, if *enni_dccstat* = ENABLED and the E-NNI node is not created by provisioning of the sigPCIP_Address in **`ENT-ULS`**.

Error code: `SNVS – Status, Not in Valid State, E-NNI Node not created`

Validation rule (UNI/E-NNI specific): Prevent that both DCCs of a port are assigned to UNI

The command is rejected, if it is tried to set *uni_dccstat* = ENABLED for a specific port and *dcctype* and for the same port *uni_dccstat* is already ENABLED for the other *dcctype*.

Error code:

`IDNV – Input, Data Not Valid, not possible to assign both channels to UNI`

Validation rule (UNI/E-NNI specific): Prevent that both DCCs of a port are assigned to E-NNI

The command is rejected, if it is tried to set *enni_dccstat* = ENABLED for a specific port and *dcctype* and for the same port *enni_dccstat* is already ENABLED for the other *dcctype*.

Error code:

`IDNV – Input, Data Not Valid, not possible to assign both channels to E-NNI`

Validation rule (UNI/E-NNI specific): Maximum number of channels assigned to UNI

The uninode can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for uninode (i.e. *uni_dccstat* = ENABLED) is not exceeded.

Error code:

`SSRE – Status, System Resources Exceeded, allowed limit of DCC channels exceeded`

Validation rule (UNI/E-NNI specific): Maximum number of channels assigned to E-NNI

The enninode can support up to 64 DCC interfaces. It is checked that the max number of 64 DCC interfaces for enninode (i.e. *enni_dccstat* = ENABLED) is not exceeded.

Error code:

`SSRE – Status, System Resources Exceeded, allowed limit of DCC channels exceeded`

Validation rule (UNI/E-NNI specific): Can't disable DCC if associated node has manual routes via this DCC

Disabling a DCC, although the associated node has manual routes via this DCC is rejected.

Error code: `SNVS – Status, Not in Valid State, Manual routes exist`

Validation rule (UNI/E-NNI specific): ENNI node must be created

If the ENNI Node is not created (by provisioning of the *sigPCIP_Address* in ENT-ULS), then the command is denied.

Error code: `SNVS – Status, Not in Valid State, ENNI Node not created`

Related TL1 messages

- "RTRV-FECOM: Retrieve DCC related parameters" (p. 3-2407)
- "RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection" (p. 3-2329)
- "ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
- "DLT-CRS-FECOM: Delete transparent DCC cross-connection" (p. 3-156)

ENT-FECOM notification

Output format

Note: In contrast to RTRV-FECOM this notification will also be sent when dccstat=DISABLE, which is the default value after creation of a DCC channel.

The **ENT-FECOM** notification has the following specific syntax:

ENT-FECOM::*port*,*dcctype*:*spec_block*

Output parameters

Command parameters

1. **port**
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID.
   Range: SPortAID; the keyword **all** is not allowed.
2. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC of the *port* is affected by the command.
   Type: Alphanumeric value set.
   Range:
   - LINE
   - SECTION.

   Default:
   - SECTION if dccstat is set to ENABLE
   - no default (i.e. parameter is mandatory) if *scn_dccstat* or *uni_dccstat* is ENABLE.

Spec block parameters

1. **dccprottp**
   This parameter reports the protection type of the DCC channel. 0X1 means the port is not involved in a protection group or DCC protection switching is disabled. 1+1 means DCC protection switching is enabled: In these case only the active worker line is shown.
   Type: Alphanumeric value set.
   Range: 0X1 (default), 1+1.
2. **dccstat**
   This parameter indicates whether communication over the DCC channel is enabled or disabled.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* must be present in the command.

3. **`lapdmode`**
   This parameter indicates the type of LAPD service. Modes are AITS and UITS. Starting with R2.0 this parameter is also used to determine the service for the LinkID protocol.
   If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set..
   Range:
   - UITS (default)
   - AITS.
4. **`lapdrl`**
   This parameter indicates the LAPD role. This parameter is also used to determine the service for the LinkID protocol. If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set.
   Range:
   - USER-SIDE (default)
   - NETWORK-SIDE.
5. **`osinode`**
   Assignment of a DCC channel to an MCN node in the NE. This parameter must be supplied when dcc_stat = ENABLED and not the default osinode1 is used, in all other cases it is omitted.
   An NE can have up to 8 instances of the OSI stack. Node-1 is default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00). Nodes 2-8 can be created and configured, see "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951).
   Type: AID.
   Range: osinode{1-8}
   Default: osinode1.

6. **scn_dccstat**
   This parameter indicates whether the DCC channel is used for communication over the SCN.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* must be present in the command.
7. **uni_dccstat**
   This parameter indicates whether the DCC channel is used for UNI communication.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat* or *enni_dccstat* must be present in the command.
8. **enni_dccstat**
   This parameter indicates whether the DCC channel is used for E-NNI communication.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Note: One and only one of the parameters *dccstat*, *scn_dccstat*, *uni_dccstat*, or *enni_dccstat* must be present in the command.

Example output

```
   NE-UNITE 00-11-03 01:00:00
A  000 REPT DBCHG
   ”TIME=01-00-00,DATE=00-12-13:ENT-FECOM::1-1-#-#-5-
      2,line:dccprottp=0X1,dccstat=DISABLE,lapdmode=UITS,osinode=osinode1”
;
```

ENT-FECOM-LAN: Enter LAN related parameters

The **ENT-FECOM-LAN** command allows manually enable or disable IP or OSI on this LAN and set related parameters. It can also be used to modify parameters when IP or OSI is enabled.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Abortable: No.

Input format

Command syntax: **ENT-FECOM_LAN**:*tid*:*aid*:*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID with one exception: the keyword **lanall** is not allowed.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.

Spec block parameters

If a LAN interface is being changed, all parameters not set explicitly in the command line will remain the actual values, except the parameter osinode, which is changed to the default value if the parameter is omitted.

1. **ip_ad** – IP Address
   This parameter indicates the assigned IP address for the LAN interface.
   The IP-address includes both, a router address and a HOST address. The HOST address that will will be used for FTP depends on the LAN interface where the related TCP connection request originates.
   Note: Setting of ip_ad comes only into effect when ip_lanstat is ENABLE and

must be consistent with ip_snm (see also validation rules).
Type: 32-bit IPv4 address: Four dot separated numeric strings.
Range: For first numeric string: 0 .. 223 For second to fourth numeric string: 0 .. 255.
The values 0.0.0.0 and 1.1.1.1 are not allowed.

2. **ip_lanstat** – IP-LAN Status
This parameter indicates whether IP communication over the external LAN channel is enabled or disabled. Note: Enabling has no effect if *lanstat* is disabled for that interface.
Type: Alphanumeric value set.
Range: ENABLE or DISABLE for LAN1, LAN2, LAN3
Default: DISABLE for LAN1, LAN2, LAN3.

3. **ip_snm** – IP Mask
Note: Setting of *ip_snm* comes only into effect when *ip_lanstat* is ENABLE and must be consistent with *ip_ad* (see also validation rules).
This parameter indicates the assigned IP subnet mask (32-bit IPv4) for the LAN interface. The full IP-address is needed as host-address. Each LAN needs its own IP-address and IP-mask, indicating that all IP-addresses that match the prefix (Unmasked part) are grouped as the subnet-ID for this LAN-subnet.
If the LAN interface is used for UNI, then the parameter ip_snm is used to define a subnet for multiple UNI-Cs which belong to the same customer.
Type: 32-bit IPv4 address: Four dot separated numeric strings or empty string. The empty string value is used as indication for an invalid/undefined IP subnet mask.
Range: For each numeric string: 0 .. 255.
Default: For LAN1, LAN2 and LAN3: empty string.

4. **isis_stat** – (I) IS-IS Status
This parameter indicates whether (integrated) IS-IS is enabled or disabled on the LAN port specified by the input *aid*.
Note: Enabling has no effect if lanstat is disabled for that interface.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE.

5. **lanrpri** – LAN Designated Router Priority
This parameter sets the relative priority for the node in the election for designed router on this LAN. There is a separate election for each separate LAN to which the node is connected. A higher value indicates a higher probability to be selected as designated router. A designated router is selected per area (between the nodes on that LAN with the same area address) and for the Level-2 nodes on the LAN.
Note: When *osinode* = scnnode modification of this parameter has no effect.
Note: When *osinode* = uninode this parameter has no effect
Note: When *osinode* = enninode this parameter has no effect.
Type: numeric string.
Range: 1 – 127.
Default: 64.

6. **`lanstat`** – LAN Status
   This parameter indicates whether physical communication over the external channel is enabled or disabled.
   Note: Disabling of an interface also affects usage of protocols over this interface, see *lan_stat* and *osi_stat*.
   Type: Alphanumeric value set.
   Range: ENABLE for lan1; ENABLE, DISABLE for lan2, lan3:
   Default: ENABLE for lan1 and lan2; DISABLE for lan3.
7. **`lantlss`** – LAN TARP Storm Suppression
   This parameter indicates the enhanced TARP propagation rules that suppress TARP LAN storms.
   Note: When *osinode* = uninode this parameter has no effect.
   Note: When *osinode* = enninode this parameter has no effect.
   Type: Alphanumeric value set.

   Range:

   - ENABLE (default)
   - DISABLE.
8. **`osinode`**
   The parameter *osinode* is always considered in the command. This means, that the parameter will be changed to the default value if it is not specified.
   Assignment of LAN port to an OSI-node, the SCN node, the UNI node, or the E-NNI node in the NE. A UNITE NE can have up to 8 instances of the OSI stack, has one SCN node, one UNI node, and one E-NNI node.
   Node-1 is the default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00).
   Node 1 is always present. The SCN node is created when the ASTN IP address is configured, the UNI node is created when the UNI IP address is configured, the SC node (ENNI node) is created when the SC IP address and the SC ID is configured. Node 2-8 can be created and configured, see command **`ENT-ULSDCC-L3`**.
   Type: AID.
   Range: For LAN1: osinode1 (fixed value).
   For LAN2 and LAN3: osinode{1-8} | scnnode | uninode | enninode.
   Note: If this parameter is set to scnnode or uninode or enninode, then *lanstat* and ip_lanstat must be ENABLED.
9. **`osi_lanstat`** – OSI LAN Status
   This parameter indicates whether OSI communication over the external channel is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE for LAN1, LAN2, LAN3

Output format

If the ENT-FECOM-LAN is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command set the condition of a LAN interface respectively or modify concerning parameters. Result should be an update of the parameters for the specified LAN interface as set in the spec_block of the command line.

The update is reported by means of ENT-FECOM-LAN notification.

Modification of the parameters *ip_ad*, *ip_snm*, *lantlss*, *isis_stat* leads to a reset of the DCF controller.

Setting lanstat to disabled shall implicitely set the following parameters to disabled:

- *osi_lanstat*
- *ip_lanstat*.

as changing *isis_stat* results in a reboot of the DCF there is no implicit change of this attribute.

After a LAN port has been assigned to the UNI node (*osinode* = uninode), the manual route entries for the LAN port is empty.

Example input/output

The following example shows the result of **ENT-FECOM-LAN**:

```
ENT-FECOM-LAN:NE-UNITE:1-1-lan2:123456:::ip_lanstat=enable,ip_ad=135.248.206.64,ip_
      snm=255.255.0.0,lanstat=enable,osinode=osinode8;
   NE-UNITE 00-12-23 11:59:00
M  123456 COMPLD
;
```

The following example shows the result of **ENT-FECOM-LAN**:

```
ENT-FECOM-LAN:NE-UNITE:1-1-lan2:123456:::ip_lanstat=enable,ip_ad=135.248.206.64,ip_
      snm=255.255.0.0,lanstat=enable,osinode=enninode;
   NE-UNITE 00-12-23 11:59:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-FECOM-LAN** command.

Validation rule: parameter *osinode*

The command is rejected when a OSI node is not in a created state. This means a value for parameter osinode is used in the command, which is different from osinode1 or scnnode and was not used for creation of a node via ENT-ULSDCC-L3 before.

Error code: `ENEQ – Equipage, Not EQuipped, osinode`

Validation rule: Can’t disable LAN1

If it is tried to set *lanstat*, or *osi_lanstat* to DISABLE for LAN1, the command will be rejected.

Error code: `SNVS – Status, Not in Valid State, LAN1 can not be disabled`

Validation rule: Can’t enable *lanstat* when *ip_lanstat* and *osi_lanstat* are DISABLED

If it is tried to set *lanstat* = ENABLE for a specific LAN, while both *osi_lanstat* and *ip_lanstat* are set to DISABLE for the same LAN, the command will be rejected.

Error code: `IDNC – Input, Data, Not Consistent`

Validation rule: Can’t enable *ip_lanstat* and/or *osi_lanstat* when lanstat is disabled

If it is tried to set osi_lanstat=ENABLE and/or ip_lanstat=ENABLE without setting lanstat=ENABLE and *lanstat* is currently disabled, then the command will be rejected.

Error code: `IDNC – Input, Data, Not Consistent`

Validation rule: Check IP configuration

If it is tried to set *ip_lanstat* to ENABLE for a LAN (LAN1, LAN2, LAN3) without a valid IP address *ip_ad* and subnetwork mask *ip_snm*, the command is denied. Valid means:

- any 32 bit IPv4 address in the range of 0.0.0.0 to 255.255.255.255.

Error code: `SNVS – Status, Not in Valid State, invalid IP configuration`

The validation rule does not cover the configuration rules for IP networks. These rules may be different for each network. In many networks the following rules are applicable:

- ip_ad must not be a multicast group address or reserved special case address, i.e. in the range of 0.0.0.0 to 223.255.255.255.
- ip_snm is in a format of one or more ’1’ bits, followed by zero or more ’0’ bits (i.e. all ’1’ are near the most significant bit, all ’0’ are near the least significant bit).

Additionally the following constraints between ip_ad and ip_snm must be true:

- (ip_ad AND ip_snm) is not equal to 0. This means that the Subnet ID is not 0.
- ((ip_ad AND ip_snm) OR NOT ip_snm) is not equal to 255.255.255.255. This means that the Subnet ID shall not consist of all 1s.
- (ip_ad AND NOT ip_snm) is not equal to 0. This means that the Host ID is not 0.
- ((ip_ad AND NOT ip_snm) OR IP_mask) is not equal to 255.255.255.255. This means that the Host ID shall not consist of all 1s.
- (ip_ad AND 255.0.0.0) is not equal to 127.0.0.0. This means that the IP address is not the (reserved) loopback address.

The AND, OR, NOT operations above are meant as bitwise logical operations on all 32 bits of *ip_ad*, *ip_snm*.

Validation rule: only *osinode1*, *scnnode, uninode* and *enninode* supports IP

If it is tried to set *ip_lanstat* to ENABLE while the LAN port is assigned to OSI node 2…8, the command is rejected.

Error code:
`SNVS – Status, Not in Valid State, Assigned Node not allowed for IP`

Validation rule: Assignmet of OsiNode without IP support

If it is tried to (re-)assign a LAN port to OsiNode2 til OsiNode8, then *ip_lanstat* must be set to disabled.

Error code:
`SNVS – Status, Not in Valid State, Assigned Node not allowed for IP`

Validation rule: only osinode1 for LAN 1

If it is tried to assign another osinode as osinode1 for LAN1, the command is rejected.

Error code:
`IDNV – Input Data Not Valid, osinode cannot be changed for LAN1`

Validation rule: all or no traffic for SCN node

For the SCN node it is only allowed to have all – lanstat, ip_lanstat, osi_lanstat and isis_stat – either enabled or disabled. Any attempt to set (any combination of) lanstat, ip_lanstat, osi_lanstat or isis_stat that would result in a violation of this rule is rejected with.

Error code: `IDNC – Input, Data Not Consistent`

Validation rule: can't enable *isis_lanstat* without enabling *osi_lanstat*

If it is tried to set isis_lanstat to enabled without also enabling osi_lanstat while osi_lanstat is disabled , the command is rejected with.

Error code: `IDNC – Input, Data Not Consistent`

Validation rule: check of the maintenance condition

The system must be in maintenance condition (MCOND=YES) for change of:

- *ip_ad*
- *ip_snm*
- *ip_lanstat*
- *osinode*
- *isis_stat*
- *osi_lanstat*
- *lanstat*
- *lantlss*
- *lanrpri*

Note: *lanpri* is independent of the maintenance condition.

Error code:

`SNVS – Status, Not in Valid State, Not in Maintenance Condition`

Validation rule: IP disabling while T-TD or FTTD is enabled

If it is tried to set *ip_lanstat* to DISABLE on all LAN ports while either *tl1tcpipgw* or *ftamftpgw* is set to ENABLE, the command is rejected.

Error code:

```
SNVS – Status, Not in Valid State, cannot disable IP on all LAN ports
while T-TD or FTTD is enabled
```

Validation rule: Can't assign LAN to *scnnode* when no ASTN IP address is provisioned

Setting *osinode* = scnnode is only allowed when a valid ASTN IP address is provisioned (see ip_ad_astn in ENT/RTRV-ULS).

Error code:

`SNVS – Status, Not in Valid State, no SCN IP address provisioned`

The necessary pre-condition that the SwiSS application is enabled is checked.

Validation rule: Can't assign LAN to *uninode* or *enninode* when no ASTN IP address is provisioned

Setting *uninode* = ENABLED or setting *enninode* = ENABLED is only allowed when a valid ASTN IP address is provisioned (see *ip_ad_astn* in ENT/RTRV-ULS).

Error code:

`SNVS – Status, Not in Valid State, no SCN IP address provisioned`

The necessary pre-condition that the SwiSS application is enabled is checked.

Validation rule: Can’t assign LAN to SCN Node when T-TD or FTTD is enabled and no other LAN port with IP configuration is available

Setting *osinode* = scnnode for LAN2 or LAN3 while T-TD or FTTD is enabled is not allowed if there is no other LAN port with valid IP configuration which is assigned to *osinode1*.

Error code:
`SNVS – Status, Not in Valid State, no other LAN port with IP enabled for T-TD or FTTD available`

Validation rule: Can’t assign same IP address to several LANs

Setting the IP address of a LAN is rejected if this IP address is already assigned to another LAN of the NE.

Error code: `IDNC – Input, Data, Not Consistent`

Validation rule: IP address already used for an active ES64U pack

Setting the IP address of a LAN port is rejected if this IP address is already assigned to an active ES64U pack in the system. An ES64U pack shall be considered as active if its switch capacity is >0.

Error code: `IDNV – Input, Data Not Valid, IP Address already used`

Validation rule: Check for IP default router LAN

Assigning a LAN to SCN node is rejected if this LAN is used as IP default router LAN.

Error code: `SNVS – Status, Not in Valid State, wrong IP configuration`

Validation rule: IP disabling while used as IP Default Router LAN

Setting *ip_lanstat* to DISABLE is rejected if this LAN is used as IP default router LAN (see RTRV-ULS).

Error code:
`SNVS – Status, Not in Valid State, LAN is in use for Default Router`

Validation rule: IP disabling while manual MCN routes exist

If it is tried to set *ip_lanstat* to DISABLE while there are still manual MCN routes provisioned for the LAN port, the the command is denied.

Error code:
`SNVS – Status, Not in Valid State, Manual MCN routes still exist`

Validation rule: Assignment to SCN/UNI/ENNI Node while manual MCN routes exist

If it is tried to assign the LAN port to the SCN Node or the UNI Node or the ENNI Node while there are still manual MCN routes provisioned for the LAN port, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, Manual MCN routes still exist`

Validation rule: *isis_stat* must be DISABLE for uninode and ennimode

When parameter *osinode* has the value *uninode* or *ennimode*, parameter *isis_stat* must have the fixed value DISABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: *lanstat* must be ENABLE for uninode and ennimode

When parameter *osinode* has the value *uninode* or *ennimode*, parameter *lanstat* must have the fixed value ENABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: *ip_lanstat* must be ENABLE for uninode and ennimode

When parameter *osinode* has the value *uninode* or *ennimode*, parameter *ip_lanstat* must have the fixed value ENABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: *osi_lanstat* must be ENABLE for uninode

When parameter *osinode* has the value *uninode*, parameter *osi_lanstat* must have the fixed value ENABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: *osi_lanstat* must be ENABLE for enninode

When parameter *osinode* has the value *enninode*, parameter *osi_lanstat* must have the fixed value ENABLE.

Error code: `SNVS – Status, Not in Valid State`

Validation rule: Can’t assign LAN to uninode when no ASTN and UNI IP address is provisioned

Setting *osinode* = uninode is only allowed when valid ASTN and UNI IP addresses are provisioned (see *ip_ad_astn* and *ip_ad_uni* in ENT/RTRV-ULS) and the SwiSS application is enabled (*xcsetup* = NN in ED/RTRV-EQPT).

Error code:
`SNVS – Status, Not in Valid State, no SCN IP address provisioned`

Validation rule: Can’t assign LAN to enninode when no ASTN and E-NNI IP address is provisioned

Setting *osinode* = enninode is only allowed when valid ASTN and E-NNI IP addresses are provisioned (see *ip_ad_astn* and *sigPCIP_Address* in ENT/RTRV-ULS) and the SwiSS application is enabled (*xcsetup* = NN in ED/RTRV-EQPT).

Error code:

`SNVS – Status, Not in Valid State, no SCN/ENNI IP address provisioned`

Validation rule: Can’t assign LAN to UNI Node or E-NNI Node when T-TD or FTTD is enabled and no other LAN port with IP configuration is available

Setting *osinode* to uninode or enninode for LAN2 or LAN3 while T-TD or FTTD is enabled is not allowed if there is no other LAN port with valid IP configuration which is assigned to osinode1.

Error code:

`SNVS – Status, Not in Valid State, no other LAN port with IP enabled for T-TD or FTTD available`

Validation rule: Can’t de-assign LAN from a node if manual routes exist

Setting *osinode* away from a node, although the node has manual routes via the LAN is rejected.

Error code: `SNVS – Status, Not in Valid State, Manual routes exist`

Related TL1 messages

- “RTRV-FECOM-LAN: Retrieve LAN channel related parameters” (p. 3-2413)

ENT-FECOM-LAN notification

Output format

The ENT-FECOM-LAN notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: The ENT-FECOM-LAN notification has the following specific syntax:

ENT-FECOM-LAN:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID with one exception: the keyword **lanall** is not allowed.

Spec block parameters

1. **ip_ad** – IP Address
   This parameter indicates the assigned IP address for the LAN interface.
   The IP-address includes both, a router address and a HOST address. The HOST address that will will be used for FTP depends on the LAN interface where the related TCP connection request originates.
   Note: Setting of ip_ad comes only into effect when ip_lanstat is ENABLE and must be consistent with ip_snm (see also validation rules).
   Type: 32-bit IPv4 address: Four dot separated numeric strings.
   Range: For first numeric string: 0 .. 223 For second to fourth numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_lanstat** – IP-LAN Status
   This parameter indicates whether IP communication over the external LAN channel is enabled or disabled. Note: Enabling has no effect if *lanstat* is disabled for that interface.
   Type: Alphanumeric value set.
   Range: ENABLE or DISABLE for LAN1, LAN2, LAN3
   Default: DISABLE for LAN1, LAN2, LAN3.
3. **ip_snm** – IP Mask
   This parameter indicates the assigned IP subnet mask (32-bit IPv4) for the LAN interface. The full IP-address is needed as host-address. Each LAN needs its own IP-address and IP-mask, indicating that all IP-addresses that match the prefix (Unmasked part) are grouped as the subnet-ID for this LAN-subnet.
   If the LAN interface is used for UNI, then the parameter *ip_snm* is used to define a

subnet for multiple UNI-Cs which belong to the same customer.
Type: 32-bit IPv4 address: Four dot separated numeric strings or empty string or empty string
Range: For each numeric string: 0 .. 255.
Default: For LAN1, LAN2 and LAN3: empty string.

4. **`isis_stat`** – (I) IS-IS Status
   This parameter indicates whether (integrated) IS-IS is enabled or disabled on the LAN port specified by the input *aid*.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE.

5. **`lanrpri`** – LAN Designated Router Priority
   This parameter sets the relative priority for the node in the election for designed router on this LAN. There is a separate election for each separate LAN to which the node is connected. A higher value indicates a higher probability to be selected as designated router. A designated router is selected per area (between the nodes on that LAN with the same area address) and for the Level-2 nodes on the LAN.
   Note: When *osinode* = scnnode modification of this parameter has no effect.
   Note: When *osinode* = uninode this parameter has no effect.
   Note: When *osinode* = enninode this parameter has no effect.
   Type: numeric string.
   Range: 1 – 127.
   Default: 64.

6. **`lanstat`** – LAN Status
   This parameter indicates whether physical communication over the external channel is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE for lan1; ENABLE, DISABLE for lan2, lan3:
   Default: ENABLE for lan1 and lan2; DISABLE for lan3.

7. **`osinode`**
   Assignment of LAN port to an OSI-node, the SCN node, the UNI node, or the E-NNI node in the NE. A UNITE NE can have up to 8 instances of the OSI stack, has one SCN node, one UNI node, and one E-NNI node.
   Node-1 is the default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00).
   Node 1 is always present. The SCN node is created when the ASTN IP address is configured, the UNI node is created when the UNI IP address is configured, the SC node (ENNI node) is created when the SC IP address and the SC ID is configured. Node 2-8 can be created and configured, see command **`ENT-ULSDCC-L3`**.

Type: AID.
Range: For LAN1: osinode1 (fixed value).
For LAN2 and LAN3: osinode{1-8} | scnnode | uninode | enninode.
Note: If this parameter is set to *scnnode* or *uninode* or *enninode*, then *lanstat* and *ip_lanstat* must be ENABLED.

8. **`osi_lanstat`** – OSI LAN Status
   This parameter indicates whether OSI communication over the external channel is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE for LAN1, LAN2, LAN3

Example output

```
   NE-UNITE 00-11-03 01:59:00
A  000 REPT DBCHG
   ”TIME=02-00-59,DATE=00-12-23:ENT-FECOM-LAN:1-1-lan2::ip_lanstat=ENABLE,ip_
      ad=135.248.206.64,ip_
      snm=255.255.0.0,lanstat=ENABLE,lanrpri=64,lantlss=ENABLE,osinode=osinode5”
;
```

ENT-IP-MAP: Enter TCP/IP map

The **ENT-IP-MAP** command can be used to create entries in the IP map.

The **ENT-IP-MAP** command allows to enter a manager IP address and its application context ID to provide T-TD access to the NE via one of the LANs assigned to node 1. To map multiple managers the command has to be executed for each manager. The *tl1tcpipgw* parameter in the **ED-NE** command enables or disables the mapping feature.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-IP-MAP**:*tid::ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.

Spec block parameters

1. **ip** – IP Address
   This parameter is used to set or retrieve the IP address(es) of the manager(s) of the NE.
   Type: IPv4 address: four dot separated numeric strings.
   Values: IPv4 address: Each numeric string: 0 .. 255. Number of entries: Mmaximum of 20 entries allowed.
   Default: null.
2. **acid** – Application Context ID
   This parameter is used to set or retrieve the application context ID to be assigned to a particular manager IP address.
   Type: Alphanumeric string of length 0 .. 23.
   Range: TL1MAINTENANCE, TL1MEMORYADMINISTRATION, TL1TEST, TL1OTHER1, TL1PEERCOMM.
   Default: TL1PEERCOMM.
   Although acid is a mandatory parameter it will not be evaluated in the NE. The parameter is present for backward compatibility.

Output format

If the ENT-IP-MAP request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command shall set the parameters as specified for the input spec_block parameters.

Example input/output

The following example shows the result of an ENT-IP-MAP command:

```
ENT-IP-MAP:LT-UNITE::123456:::acid=TL1MEMORYADMINISTRATION,
ip=198.78.46.8;

   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ENT-IP-MAP** command.

Validation rule: Check for correct IP configuration

The adding of an entry to the IP map is only allowed if there is already a valid IP configuration (ip_ad – IP address of LAN, ip_snm – subnet mask of LAN) provisioned for at least one LAN port.

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

Validation rule: Check for duplicate entries

The adding of an entry to the IP map is not allowed, if the IP address is already contained in one of the existing entries.

Error code: `IDNV – Input, Data Not Valid, duplicate IP Map entry`

Validation rule: Check if limit of 20 IP MAP entries exceeded

The adding of an entry to the IP map is not allowed, if the IP map contains already 20 entries.

Error code:
`SSRE – Status, System Resources Exceeded, limit of IP-MAP entries exceeded`

Related TL1 messages

- “RTRV-IP-MAP: Retrieve TCP/IP map” (p. 3-2436)
- “DLT-IP-MAP: Delete TCP/IP map” (p. 3-198)

ENT-IP-MAP notification

Output format

The **ENT-IP-MAP** command shall not generate a REPT DBCHG notification. This is due to a potential security hazard: any listening manager who receives this notification could otherwise fake its own IP address to get access to the T-TD.

ENT-LONIM notification

Output format

Command syntax:

ENT-LONIM-*modifier*:*aid*::*spec_block*,

Output parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the low-order tributary. The purpose is to distinguish between: – SDH/SONET tributaries.
   Type: Alphanumeric value set.
   Range:
   - LOVC
   - LOVT
   - DS1
2. **aid** – Access Identifier
   The access identifier specifies a low-order tributary to be operated on.
   Type: AID
   Range:
   - GenLOTribAID
   - DS1NumAID

   The following values are not used:
   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID
   - *all* in Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
   - *all* in Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID

   No wildcards are allowed.

Spec Block Parameters

1. **crslpbkstat** – Cross-connect Loopback Status
   This parameter shall report the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).
   The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- YES: for enabled loopback
- NO: for disabled loopback.

Default: NO.

2. **degthr** – Signal Degrade Threshold
This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (i.e. in contrast to RTRV-VCGTRIB, RTRV-VC3, RTRV-STS1).
The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range: –5, –6, –7, –8, –9; that means, the range is 10^{-9} to 10_{-5}.
Default: –6.
3. **dexcthr** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
Default: –3.
4. **explptrc** – Expected LO Path Trace Identifier
This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Hexadecimal number without a preceding “H”.
Range: length: 32 hexadecimal; digits: value specific string acc. ITU/ETSI mode 1
Default: 89000000000000000000000000000000
5. **feppm** – Far End Path PM Enable
This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set
Range: DISABLE (default), ENABLE.

6. **`fmt`** – Digital Signal Format for DS1
   This parameter is used to specify the digital signal format expected to be received from the DS1 at the specified port(s). The TransMUX circuit pack supports DS1 path PM; it does not support DS1 line PM.
   The parameter is only supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - SF: Super Frame, supports near-end DS1 path and line PM only.
   - ESF: Extended Super Frame, supports near-end and far-end DS1 path and line PM.

   Default: ESF.
7. **`lonimpn`** – LO NIM ASAP Profile Name
   This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
   The ASAP type depends on the *modifier*:
   - for LOVC/LOVT modifier, an ASAP of type PATH is required
   - for DS1 modifier, an ASAP of type DS1 is required

   Type: Alphanumeric string. A hyphen (“ – -”) is allowed in the name additionally.
   Range: 1 to 24 characters
   Default: DEFAULT.
8. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format ptrcrfmt is set to 16 (16-byte specific-string mode).
   This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE.
   Default: DISABLE.
9. **`neppm`** – Near End Path PM Enable
   This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.

10. **`outpm`** – Outgoing Near End Path PM Enable
The parameter *outpm* specifies whether the performance monitoring for the near end in outgoing direction is enabled or disabled for this tributary.

The parameter is only supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE
Default: DISABLE.

11. **`pmlpfeses`** – PM LO Path Far End SES Threshold
This parameter specifies the Low Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end.
The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range:

The range depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF:

- 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
- 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
- 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary)

Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary).

12. **`pmlpfeses`** – PM LO Path Far End SES Threshold
    This parameter specifies the Low Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end.
    The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Integer.
    Range:

    The range depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF:

    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
    - 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
    - 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary)
    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a DS1 tributary)

    Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary), 1544 (if the entity represents a DS1 tributary).
13. **`ptca`** – Path TCA Profile Pointer
    The parameter *ptca* specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    The TCA Profile type depends on the modifier.

    - for LOVC *modifier*, a TCA Profile of type LOVC is required
    - for LOVT *modifier*, a TCA Profile of type LOPATH is required
    - for DS1 *modifier*, a TCA Profile of type DS1 is required

    Type: Alphanumeric string.
    Range: 1 … 24 characters.
    Default: DEFAULT0.
14. **`ptrcrfmt`** – Path Trace Read Format
    This parameter specifies the format of the path trace (J1 or J2 byte) to be received (sink).

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Numeric string.

Range:

- 2 (default), for non-specific repeating byte
- 16, for 16-byte specific string
- 64, for 64-byte string (only applicable if the entity represents a LO-VC3 tributary).

To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.

15. **tcamd** – TCA mode

    This parameter is only available for SDH mode and will therefore only be reported if the output parameter *aidtype* is LOVC. The parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable for SDH:

    - TR-only. (also known as the implicit reset method)
    - TR-RTR. (also known as the explicit reset method).

    For SONET the parameter *tcamd* is not supported. SONET ports only support "TR-only" mode. Thus the mode must be internally set to ″"TR-only".

    The parameter is not supported for the following modifiers:

    - DS1
    - LOVT

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: TR, TRRTR

    Default:

    - SDH mode: TRRTR
    - SONET mode: TR (internal).

    For SONET the parameter *tcamd* is not supported. SONET ports only support "TR-only" mode. Thus the mode must be internally set to ″"TR-only".

16. **tmonmd** – Tributary Monitoring Mode

    This parameter specifies the monitoring mode for the tributary.
    Type: Alphanumeric value set.
    Range: MON, NMON.
    Default: NMON.

Example output

Example of an **ENT-LONIM** notification when creating a SDH substructure:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ENT-LONIM-LOVC:\
    1-1-#-#-39-16-1-1&&-63::\
    crslbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lptimmon=DISABLE,lonimpn=DEFAULT,neppm=DISABLE,pmlpfeses=600,\
```

```
    pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=1,tcamd=TRRTR,tmonmd=NMON"
;
```

Example of an **ENT-LONIM** notification when creating a SONET substructure:

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:ENT-LONIM-LOVT:1-1-#-#-39-16-3-1&&-28::\
    crslbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=890000000000000000000000000000000,feppm=DISABLE,\
    lptimmon=DISABLE,lonimpn=DEFAULT,neppm=DISABLE,pmlpfeses=600,\
    pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=1,tmonmd=NMON"
;
```

Example of an **ENT-LONIM** notification for creation of DS1 tributaries at a TransMUX port:

```
   UNITE-NE 08-03-01 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-50,DATE=08-03-01:ENT-LONIM-DS1:\
    1-1-#-#-15-tma1-1-d1&&-d28::fmt=SF,lonimpn=DEFAULT,\
    outpm=ENABLE,pmlpneses=1544,ptca=DEFAULT0,tmonmd=NMON"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-LONIM** command, too.

Validation rule: Check LONIM::modifier against PTF::istd

If the command is applied with a modifier which does not match to the interface standard parameter istd of the owning PTF entity then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Following combinations are allowed: *modifier istd*:

LOVC, SDH LOVT, SONET ALL, SDH or SONET.

Validation rule: Check *aid* and *dyn* consistency

If the command is applied with an aid parameter containing wildcards (all at any position) and the dyn parameter is set to YES then the command is denied.

Error code: `IDRG – Input, Data Range Error`

Example: Given the aid refers to 1-1-#-#-39-16-1-all and the dyn parameter is set to YES then the command will be rejected.

Related TL1 messages

- "ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)

ENT-LOPTF notification

Output format

Syntax: **ENT-LOPTF**:-*modifier*:*aid*::*spec_block*

Output parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the low-order tributary.
   Type: Alphanumeric value set.
   Range: TMSVT.
2. **aid** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: TMuxSLONumAID.

   The following values are not used:

   - **all**, **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID
   - **all** for Simple_TMuxSPTNumAID
   - **all** for Simple_TMuxSLONumAID

   No wildcards are allowed.
   Default: None.

Spec block parameters

1. **degthr** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range (as exponents of 10): –5; –6; –7; –8; –9, that means, the range is 10^{-9} to 10^{-5}.
   Default: –6.
2. **dexcthr** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **RTRV-VCGTRIB**, **RTRV-VC3**, **RTRV-STS1**).

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
Default: –3.

3. **`explptrc`** – Expected LO Path Trace Identifier
This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding "H".
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000

4. **`lopn`** – LO ASA Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID. For *modifier* = TMSVT, an ASAP profile of type *PATHTERM* is required.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: 1 to 24 characters.
Default: DEFAULT.

5. **`lptimais`** – LO Trace Identifier Mismatch AIS Insertion
This parameter specifies whether a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
Type: Alphanumeric value set.

   Range:

   - DISABLE
   - ENABLE

   Default: The default value depends on interface standard:

   - SONET: DISABLE.

6. **`lptimmon`** – LO Path Trace Identifier Mismatch Monitor
This parameter specifies the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).

   This parameter is not supported for the following modifiers:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- DISABLE
- ENABLE

Default: DISABLE

7. **`ptrcrfmt`** – Path Trace Read Format
   The parameter *ptrcrfmt* specifies the format of the path trace (J1 or J2 byte) to be received (sink).
   The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Numeric string.
   Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string
   - 64: for 64-byte string.

   To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.
   Default: 2.
8. **`ptrcwfmt`** – Path Trace Write Format
   This parameter specifies the format of the path trace (J2 byte) to be sent (source).
   Type: Numeric string.
   Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string

   Default: 2.
9. **`tmonmd`** – Tributary Monitoring Mode
   This parameter specifies the monitoring mode for the tributary.
   Type: Alphanumeric value set.
   Range:
   - MON
   - NMON

   Default: NMON.
10. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J2) to be sent (source). For the 16-byte path trace format, the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    Type: Hexadecimal number without a preceding “H”.

Range: The range depends on the parameter *ptrcwfmt* specified in **RTRV-PTF** output parameters

- length: 32 hexadecimal digits.
- value: specific string acc. ITU/ETSI mode 1.

Default: 00.

Example input/output

Example of an **ENT-LOPTF** notification:

```
   UNITE-NE 08-03-01 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-50,DATE=08-03-01:ENT-LOPTF-TMSVT:1-1-#-#-39-tms16-1-1&&-28::\
    degthr=-6,dexcthr=-3,explptrc=89000000000000000000000000000000,\
    lopn=DEFAULT,lptimais=DISABLE,lptimmon=DISABLE,\
    ptrcrfmt=2,ptrcwfmt=2,tmonmd=NMON,trc=00″
;
```

ENT-MANROUTE-LAN: Enter manual route at LAN port

The **ENT-MANROUTE_LAN** command is used to enter manual routes at LAN ports, which are used as UNI control channels.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-MANROUTE_LAN**:*tid*:*aid*:*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*.

Spec block parameters

1. **ip_ad** – destination IP Address
   This parameter is used to indicate the destination IP address to which a manual route is created. This is the LAN i/f address of UNI-C
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – Node ID of UNI-C
   This parameter is used to indicate the router ID of the UNI-C to which a manual route is created.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255. The values 0.0.0.0 and 1.1.1.1 are not allowed.

3. **`ip_snm`** – IP Mask
   This parameter is used to indicate the subnet mask of the destination IP address to which a manual route is created.
   Type: Four dot separated numeric strings
   Range: For each numeric string: 0 .. 255.
4. **`osinode`**
   This parameter is used toindicate the OSI node assigned to the LAN port.
   Type: AID.
   Range: uninode.

Output format

If the **`ENT-MANROUTE-LAN`** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **`ENT-MANROUTE-LAN`** command shall add a manual route entry for a LAN port, which is used as control channel for UNI.

The update of the manual routes are notified with ENT-MANROUTE-LAN notifications.

Example input/output

The following example shows the result of ENT-MANROUTE-LAN:

```
ENT-MANROUTE-LAN:LT-UNITE:1-1-lan2:123456:::ip_ad=135.168.90.4,ip_ad_
      uni=135.168.90.4,ip_snm=255.255.0.0,osinode=uninode;

   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ENT-FECOM-LAN`** command.

Validation rule: Not allowed to use LAN1 for UNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for UNI`

Validation rule: LAN port not used for UNI

If the LANport specified in the AID is not in use for UNI, i.e. is not assigned to the uninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, LAN port not in use for UNI`

Validation rule: No duplicate Manual Route entries

The command is denied, if a manual route entry regardless of the LAN port already exists which has the same values in the parameters *ip_ad* and *ip_ad_uni*.

Error code:
`IDNV – Input, Data Not Valid, Manual route entry already exists`

Validation rule: Check if limit of 20 Manual Routes per LAN port exceeded

The addition of a manual route entry for a LAN port is denied, if already 20 manual route entries exist for the LAN port.

Error code:
`SSRE – Status, System Resources Exceeded, limit of manual route entries exceeded`

Related TL1 messages

- "RTRV-MANROUTE-LAN: Retrieve manual route at LAN port" (p. 3-2516)
- "DLT-MANROUTE-LAN: Delete manual route at LAN port" (p. 3-203)

ENT-MANROUTE-LAN notification

Output format

Command syntax: **ENT-MANROUTE_LAN**:*aid*:::*spec_block*;

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.

Spec block parameters

1. **ip_ad** – destination IP Address
   This parameter is used to indicate the destination IP address to which a manual route is created. This is the LAN i/f address of UNI-C
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – Node ID of UNI-C
   This parameter is used to indicate the router ID of the UNI-C to which a manual route is created.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255. The values 0.0.0.0 and 1.1.1.1 are not allowed.
3. **ip_snm** – IP Mask
   This parameter is used to indicate the subnet mask of the destination IP address to which a manual route is created.
   Type: Four dot separated numeric strings
   Range: For each numeric string: 0 .. 255.
4. **osinode**
   This parameter is used toindicate the OSI node assigned to the LAN port.
   Type: AID.
   Range: uninode.

Example output

The following example shows the result of **ENT-MANROUTE-LAN**:

```
   NE-UNITE 06-06-30 08:59:00
A  000 REPT DBCHG
   ”TIME=08-59-00,DATE=06-06-30:ENT-MANROUTE-LAN:1-1-lan2::ip_ad=135.248.206.64, ip_
      ad_uni=135.248.206.64,ip_snm=255.255.0.0, osinode=uninode”
;
```

ENT-MANROUTE-MCN: Enter manual IP route for management communication

The **ENT-MANROUTE_MCN** command is used to enter manual IP routes for management communication.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: Yes.

Input format

Command syntax: **ENT-MANROUTE_MCN**:*tid*:*aid*:*ctag*:::[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   The keyword "lanall" is allowed for Simple_LANPortAID.
   Default: Omitting this field has the same effect as the keyword "lanall", that means, the report will display the manual MCN route entries of all possible LAN ports assigned to the MCN.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`dest_ipad`** – Destination IP Address
   The IP address of an IP node which can be reached via one of the LAN ports.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: None.
2. **`dest_ipmask`** – Destination IP Subnet Mask
   The IP subnet mask of the IP node which can be reached via the LAN port.
   Type: Four dot separated numeric strings
   Range: Each numeric string: 0 … 255.
   Default: None.
3. **`nexthop_ipad`** – Next Hop IP Address
   The IP address of the next hop routing node via which the Destination IP Address can be reached.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The value 0.0.0.0 is used if the IP node defined by *dest_ipad* is connected to the LAN port directly. The values 1.1.1.1 and 127.0.0.1 are not allowed.
   Default: None.

Output format

If the **`ENT-MANROUTE-MCN`** request is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`ENT-MANROUTE-MCN`** command adds a manual MCN IP route entry for the system. The update of the manual routes is notified with **`ENT-MANROUTE-MCN`** notifications.

Example input/output

The following example shows the result of an **`ENT-MANROUTE-MCN`** command:

```
ENT-MANROUTE-MCN:UNITE-NE:1-1-lan2:123456:::dest_ipad=192.168.90.4,dest_
      ipmask=255.255.255.0,nexthop_ipad=192.168.2.45;
```

```
   UNITE-NE 08-06-20 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-MANROUTE-MCN** command.

Validation rule: Auto Encapsulation must be enabled

The command is denied, if the Auto Encapsulation function is not enabled.

Error code:

`SNVS – Status, Not in Valid State, Auto Encapsulation not enabled`

Validation rule: Only osinode1 supports Auto Encapsulation

The command is denied, if the specified LAN port is not assigned to osinode1.

Error code:

`IDNC – Input, Data Not Consistent, Assigned Node not allowed for Auto Encapsulation`

Validation rule: IP communication must be enabled for LAN port

The command is denied, if the IP communication is not enabled for the specified LAN port (parameter *ip_lanstat* not set to ENABLE).

Error code:

`SNVS – Status, Not in Valid State, IP communication not enabled for LAN port`

Validation rule: No duplicate Manual MCN IP Route entries

The command is denied, if a manual MCN IP route entry already exists which has the same value in the parameter *dest_ipad*.

Error code:

`IDNV – Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`

Validation rule: No more than one LAN connected to same next hop

The command is denied if the provided *nexthop_ipad* is already provisioned for any other LAN interface.

Error code:

`IDNV – Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`

Validation rule: Check if limit of 20 Manual MCN IP Routes exceeded

The addition of a manual MCN IP route entry is denied, if already 20 manual route entries exist in the system.

Error code:

`SSRE – Status, System Resources Exceeded, limit of manual route entries exceeded`

Related TL1 messages

- “RTRV-MANROUTE-MCN: Retrieve manual IP route for management communication” (p. 3-2519)
- “ENT-MANROUTE-MCN notification” (p. 3-1618)
- “DLT-MANROUTE-MCN: Delete one manual MCN IP route entry” (p. 3-207)
- “DLT-MANROUTE-MCN notification” (p. 3-209)

ENT-MANROUTE-MCN notification

Output format

Command syntax: **`ENT-MANROUTE_MCN`**:*aid*:::*spec_block*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   The keyword “lanall” is allowed for Simple_LANPortAID.
   Default: Omitting this field has the same effect as the keyword “lanall”, that means, the report will display the manual MCN route entries of all possible LAN ports assigned to the MCN.

Spec block parameters

1. **`dest_ipad`** – Destination IP Address
   The IP address of an IP node which can be reached via one of the LAN ports.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: None.
2. **`dest_ipmask`** – Destination IP Subnet Mask
   The IP subnet mask of the IP node which can be reached via the LAN port.
   Type: Four dot separated numeric strings
   Range: Each numeric string: 0 … 255.
   Default: None.
3. **`nexthop_ipad`** – Next Hop IP Address
   The IP address of the next hop routing node via which the Destination IP Address can be reached.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The value 0.0.0.0 is used if the IP node defined by *dest_ipad* is connected to the LAN port directly. The values 1.1.1.1 and 127.0.0.1 are not allowed.
   Default: None.

Example output

The following example shows the result of **`ENT-MANROUTE-MCN`**:

```
   UNITE-NE 08-06-20 08:00:02
A  000 REPT DBCHG
   ”TIME=08-00-00,DATE=08-06-20:ENT-MANROUTE-MCN:1-1-lan2::dest_
      ipad=192.168.90.4,nexthop_ipad=192.168.2.4,nexthop_ipmask=255.255.255.0”
```

;

ENT-OC3 notification

Output format

Syntax: **`ENT-OC3`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC3 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.

6. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

7. **`felpm`** – Far-end Line PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

8. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is setable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
    Default: 31.
    Example: An O-3 signal might be represented by 31 which implies 3 STS1's or 13 which implies 1 STS3C.
11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 or STS3 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default) or STS3.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID` –** Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **networkTNAAddress** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET:

    - STM1: 28800
    - OC3: 154.

28. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

29. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM1: 2400
    - OC3: 155.
31. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
32. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
33. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.
34. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

35. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
36. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
37. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: *""*
38. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.

Range:

- ENABLE: enable consequent action for TIM
- DISABLE (default): disable consequent action for TIM.

39. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
40. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
41. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
42. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
43. **`sfthr`** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

44. **sltca** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

45. **sonet** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see "RTRV-STM4: Retrieve STM4 port" (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
Default: DEFAULT.

46. **spm** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

47. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
49. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

51. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
    Type: Alphanumeric value set.
    Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
52. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
    Default: 3 1 = all STS1.
53. **`z0outmode`** – Mode of outgoing Z0 byte insertion
    The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
    Type: Alphanumeric value set.
    Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-OC3:1-1-#-#-1-1::servcond=ENABLE,
      sdsfmode=POISSON,sdthr=-
      6,brstthr=30,....,istd=SONET,inputsig=31,unequiposig=31″
;
```

ENT-OC12 notification

Output format

Syntax: **`ENT-OC12`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-12 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future. This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **crstat** – Constituent Rates Status list
This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.

6. **exprstrc** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

7. **felpm** – Far-end Line PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

8. **frcdus** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **fsflpbkstat** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
10. **inputsig** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
    Default: 121.
    Example:
    An OC12 signal might be represented by 31-33 which implies 3 STS1's, 3 STS3C's.
11. **istd** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
12. **linkResCls** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3 or STS12 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3 or STS12.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
  0000000000000000000000000000000000000000…
  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **pmmsfeses** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:

    - STM4: 115200
    - OC12: 615.

28. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

29. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000.
    Default: The default values differ for SDH and SONET and the different rates:
    - STM4: 2400
    - OC12: 616.
31. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
32. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
33. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.
34. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

35. **remLinkID** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

36. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

38. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.

Range:

- ENABLE: enable consequent action for TIM
- DISABLE (default): disable consequent action for TIM.

39. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
40. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
41. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
42. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
43. **`sfthr`** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

44. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

45. **`sonet`** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen (“-”) is allowed in the name additionally.
Default: DEFAULT.

46. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
49. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

51. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
52. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
Default: 121 = all STS1 (fixed).
53. **`z0outmode`** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ENT-OC12:1-1-#-#-1-1::servcond=ENABLE,
      sdsfmode=POISSON,sdthr=-
      6,brstthr=30,....,istd=SONET,inputsig=121,unequiposig=121"
;
```

ENT-OC48 notification

Output format

Syntax: **`ENT-OC48`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-48 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future. This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48 representing STS1, STS3C, STS12C, STS48C.

6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

7. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).

8. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **fsflpbkstat** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

10. **inputsig** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
    Default: 481.
    Example: An OC-48 signal might be represented by 91-13-112-241 which implies 9 STS1's, 1 STS3C, 1 STS12C, 24 STS1's.

11. **istd** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONET, SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

12. **linkResCls** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12 or STS48 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12 or STS48.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
    This identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is introduced specifically for use by NN application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
  0000000000000000000000000000000000000000…
  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.
28. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

29. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
30. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
31. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
32. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

33. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …255. The unit of this value is ″milliseconds″.
34. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
35. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
36. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

37. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
38. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.
39. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
40. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
41. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.

42. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
43. **`sfthr`** – Signal Failure Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
44. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
45. **`sonet`** – SONET ASAP
    This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
46. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
47. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are

set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
- 1: for specific repeating byte (SONET)
- 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

50. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.

Range:

- 1: for specific repeating byte (SONET)
- 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

51. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

52. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
Default: 481 = all STS1 (fixed).

53. **`z0outmode`** – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
```

```
   "TIME=08-00-12,DATE=00-11-28:ENT-OC48:1-1-#-#-1-1::servcond=ENABLE,
      sdsfmode=POISSON,sdthr=-
      6,brstthr=30,....,istd=SONET,inputsig=481,unequiposig=481"
;
```

ENT-OC48T notification

Output format

Syntax

`ENT-OC48T`*:aid::spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-48 port AID.

Spec block parameters

1. **`exprstrc`** – Value of Expected J0
   For ENT-OC48T Notification this parameter refers to the incoming (ingress) direction.
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

3. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

4. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET.
   Type: Alphanumeric value set.
   Range: SONET.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

7. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   For ENT-OC48T Notification this parameter refers to the incoming (ingress) direction.
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
8. **`pmrsneses`**out – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
9. **`rstimmon`** – RS TIM Monitor Enable
   For ENT-OC48T Notification this parameter refers to the incoming (ingress) direction.
   Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
   This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
   Type: Alphanumeric value set.
   Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor

10. **rstimmon**out – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor

11. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

12. **sltca** – Section and Line TCA Profile Pointer
Definition of the section and line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

13. **sonet** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (*sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

14. **spm** – Section PM Enable
For ENT-OC48T this parameter refers to the incoming (ingress) direction.
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

15. **spmout** – Section PM Enable in Outgoing Direction
Enable or disable section performance monitoring for the outgoing (egress) direction.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).

16. **strcfmt** – Format of incoming J0
    For ENT-OC48T this parameter refers to the incoming (ingress) direction.
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
17. **strcfmtout** – Format of outgoing J0
    The read format of J0 in outgoing (egress) direction, the outgoing section trace (*strcout*) and the expected section trace (*exprstrcout*).
    Type: Numeric string.
    Range:
    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ”Time=8-00-30,DATE=00-11-20:DLT-OC48T:1-1-#-#-4-1
;
```

ENT-OC192 notification

Output format

Syntax: **ENT-OC192**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of the OC192 port.
   Type: AID.
   Range: SPortAID of OC192 port.
   The keyword **all** is not allowed.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the tributary input rate signal list.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C, STS192C.

6. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

7. **`fec`** – FEC Enable
Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
Type: Alphanumeric value set.
Range:
DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics

- OC192 / STM64 client ports (related to OTN ports)
- Basic OC768 / STM256 low cost intra office (2 km).

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

Default:

DISABLE (only) for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC192 / STM64 low cost intra office 1310 nm optics
- Basic OC768 / STM256 low cost intra office (2 km).

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

8. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

10. **fsflpbkstat** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
11. **inputsig** – Tributary Input Signal Rate List
This parameter is a list of STSn rates across the port's bandwidth.
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
Default: 1921.
12. **istd** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
Type: Alphanumeric value set.
Range: SDH.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
13. **linkResCls** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

14. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
15. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
16. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

17. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12, STS48 or STS192 (default)
18. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12, STS48 or STS192.
19. **`msrei`** – MS-REI mode
    Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
    Type: Alphanumeric value set.
    Range:
    - SINGLE
    - EXTENDED (default).
20. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN

21. **neighborASTNNodeID** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **neighborNodeID** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **neighborPortID** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
26. **`nelpm`** – Near-End Line PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
27. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`och`** – optical channel enable
    Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don't have any impact on transmission.
    Type: Alphanumeric value set.
    Range:
    - for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
    - for client ports on packs which support OTN: DISABLE: disable, optical channel not applicable.

    Default:
    - for OP10 packs, which cannot support OCH: DISABLE
    - for OP40 packs, which cannot support OCH: DISABLE
    - for OP40 packs, which only support OCH: ENABLE
    - for client ports on packs which support OTN: DISABLE.
30. **`otnclient`** – OTN Client Port
    This parameter indicates whether the port is an OTN Client Port or an optical SDH/SONET port.
    Type: Alphanumeric value set.
    Range:
    - YES for OTN client port
    - NO for optical SDH/SONET port.
31. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200

- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

32. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

33. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

    Default:
    - AUTO for optical SDH/SONET ports
    - NMON for client ports.

34. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
35. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.
36. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
37. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
38. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

39. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
40. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
41. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
42. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

43. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.
44. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
45. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
46. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
47. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

48. **`sfthr`** – Signal Failure Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

49. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

50. **`sonet`** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (*sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

51. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

52. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

53. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
54. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
55. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

56. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
57. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port is allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12, 48 or 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
Default: 1921 = all STS1 (fixed).

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ENT-OC192:1-1-#-#-1-1::servcond=ENABLE,
      sdsfmode=POISSON,sdthr=-6,brstthr=30,....,istd=SONET,
inputsig=1921,unequiposig=1921"
;
```

ENT-OTU2 notification

Output format

Command syntax: **`ENT-OTU2`**:*aid::spec_block*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OTU2 port.
   Type: AID
   Range: OTNPortAID of OTU2 port.
   The keyword ″otall″ is not allowed.

Spec block parameters

1. **`expotutti`** – Expected OTU TTI
   Expected OTU trail trace identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the expected OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte hexadecimal string
   Default: 15 bytes each with value 00.
2. **`fectype`** – FEC Type
   Forward error correction type.
   Type: Alphanumeric value set.
   Range: RS_FEC, V_FEC (default).
3. **`ochpm`** – OCH PM Enable
   Enable or disable optical channel performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
4. **`opwind`** – Optical Window
   Optical window (L-Band, C-Band) supported by the port.
   Type: Alphanumeric value set.
   Range: BAND_L for L-Band, BAND_C for C-Band.
5. **`opwl`** – Optical Wavelength
   Optical Wavelength in nanometers.
   Type: numeric value set
   Range:
   - 35 wavelengths in C-Band (1554.537 … 1568.362), i.e:

1554.537, 1554.940, 1555.343, 1555.747, 1556.151, 1556.555, 1556.959, 1557.363, 1557.768, 1558.173, 1558.578, 1558.983, 1559.389, 1559.794, 1560.200, 1560.606, 1561.013, 1561.419, 1561.828, 1562.233, 1562.640, 1563.047, 1563.455, 1563.863, 1564.271, 1564.679, 1565.087, 1565.496, 1565.905, 1566.314, 1566.723, 1567.133, 1567.542, 1567.952, 1568.362

- 93 wavelengths in L-Band (1568.773 … 1607.466), i.e:

  1568.773, 1569.183, 1569.594, 1570.005, 1570.416, 1570.828, 1571.239, 1571.651, 1572.063, 1572.476, 1572.888, 1573.301, 1573.714, 1574.127, 1574.540, 1574.954, 1575.368, 1575.782, 1576.196, 1576.610, 1577.025, 1577.440, 1577.855, 1578.270, 1578.686, 1579.102, 1579.518, 1579.934, 1580.350, 1580.767, 1581.184, 1581.601, 1582.018, 1582.436, 1582.854, 1583.271, 1583.690, 1584.108, 1584.527, 1584.946, 1585.365, 1585.784, 1586.203, 1586.623, 1587.043, 1587.463, 1587.884, 1588.304, 1588.725, 1589.146, 1589.568, 1589.989, 1590.411, 1590.833, 1591.255, 1591.678, 1592.100, 1592.523, 1592.946, 1593.369, 1593.793, 1594.217, 1594.641, 1595.065, 1595.489, 1595.914, 1596.339, 1596.764, 1597.189, 1597.615, 1598.041, 1598.467, 1598.893, 1599.320, 1599.746, 1600.173, 1600.600, 1601.028, 1601.455, 1601.883, 1602.311, 1602.740, 1603.168, 1603.597, 1604.026, 1604.455, 1604.885, 1605.314, 1605.744, 1606.174, 1606.605, 1607.035, 1607.466

6. **`otnasap`** – OTN ASAP
   OTN Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type OTN to which the designated AID points.
   Type: Alphanumeric string A hyphen (“-”) is allowed in the name additionally.
   Range: DEFAULT.
7. **`otutimca`** – OTU TIM Consequent Action Enable
   OTU Trail Trace Identifier Mismatch consequent action enable. The parameter *otutimca* shall enable or disable the TIM consequent action. In addition, to the TIM consequent action also the TIM detection is enabled or disabled.
   Type: Alphanumeric value set.
   Range: ENABLE: enable consequent action for TIM (default), DISABLE: disable consequent action for TIM.
8. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default):
     indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON:
     indicates that alarm monitoring is taking place on the port.
   - NMON:
     means that alarm monitoring is not occurring on the port.

9. **`ptype`** – Port Type
   This parameter identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range: OTU2
   `txenable` – Optical Output Enable
   Optical output enabling.
   Type: Alphanumeric value set.
   Range:
   - DISABLE (default): optical output disabled
   - ENABLE: optical output enabled.
10. **`votutti`** – Value of outgoing OTU TTI
    Value of the transmitted OTU Trail Trace Identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the transmitted OTU TTI.
    Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
    Range: 15 byte SAPI (Source Access Point Identifier).
    Default: 15 bytes each with value 00.

Example input/output

```
   UNITE-NE 06-06-30 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=06-06-30:ENT-OTU2:1-1-#-#-1-1::
      expotutti=000000000000000000000000000000,fectype=V_
      FEC,ochpm=ENABLE,opwind=BAND_
      L,opwl=1569.183,otuasap=DEFAULT,otutimca=DISABLE,pmode=MON,ptype=OTU2,txenable=ENABLE,vot
;
```

ENT-OUC: Enter ONNS to UPSR/SNCP constructs

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 3

Abortable: No.

Input format

Command syntax:

ENT-OUC:*tid:wrk_aid,prt_aid:ctag::[ouc_dir][:spec_block];*

NOTE: The time between ONNS path creation (see "ENT-PATH: Create ONNS path connection" (p. 3-1701)) and executing the **ENT-OUC** command should be more than five seconds.

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **wrk_aid** – Working Tributary Access Identifier
   This parameter specifies the working tributary of the OUC.
   Type: STS1NumAID
   Range: n.a. (**all** values shall never occur as part of an output parameter)
3. **prt_aid** – Protection Tributary Access Identifier
   Protection Tributary Access Identifier. Specifies the protection tributary of the OUC.
   Type: STSNumAID
   Range: n.a. (**all** values shall never occur as part of an output parameter)
4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
5. **ouc_dir** – OUC Directionality
   Directionality of the OUC
   Type: Alphanumeric value set.

Range:

- BIDIR (default)
- UNIDIR-ADD
- UNIDIR-DROP.

Spec block parameters

1. **`htmr`** – Hold Off Timer
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine of the path protection groups created via OUC creation.
   Type: numeric string [ms]
   Range: 0 -.. 99 ms in 1 ms steps and 100 ..- 10000 ms in100 ms steps
   Default: 0
2. **`rme`** – Revertive Mode Enable
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection groups created via OUC creation. The value ENABLE is used if the protection operates in revertive mode. The value DISABLE is used if the protection operates in non-revertive mode.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
3. **`snctype`** – SNCP Type
   This parameter specifies the type of the path protection w.r.t. switch criterias used by the protection state machine of the path protection groups created via OUC creation.
   The value SNCI (Inherent) is used if server signal fail is used as switch criteria.
   The value SNCN (Non-Instrusive) is used if server signal fail and defects supervised by a non-intrusive path monitor are used as switch criteria
   The value NA is used if the tributary inputs are related to ports operating in SONET mode.
   Type: Alphanumeric value set.
   Range: SNCI, SNCN, NA
   Default:
   - SNCN, if affected tributaries are related to ports operating in SDH mode
   - NA, if affected tributaries are related to ports operating in SONET mode.
4. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode of the path protection groups created via OUC creation.
   Type: numeric string [minute]
   Range: 0 – 60.
   Default: 5

Output format

If the **ENT-OUC** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

When setting up an OUC via ENT-OUC immediately after having setup the underlying ONNS connection via ENT-PATH, there should be a waiting time of at least 5 seconds between both these commands. If there is not enough time between both commands, the **ENT-OUC** command might be denied with "SNVS – Status, Not in Valid State, No matching NN cross-connection existing on working tributary" and would need to be repeated in such a case.

Example input/output

A bidirectional OUC between working tributary 4 of a 10G SDH EDGE port in slot 01 and protection tributary 7 of 2G5 SDH EDGE port 3 in slot 34 will be established by:

```
ENT-OUC:UNITE-NE:1-1-#-#-1-1-4,1-1-#-#-34-3-7:123456::BIDIR:
htmr=3000,rme=ENABLE,snctype=SNCI,wtr=10;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

A unidirectional drop OUC between working tributary 4 of 2G5 SONET EDGE port 1 in slot 10 and protection tributary 7 of 2G5 SONET EDGE port 3 in slot 34 will be established by the following command: (note that spec block parameter *snctype* can be omitted in the SONET case and spec block parameter *wtr* can be omitted due to *rme* being DISABLE)

```
ENT-OUC:UNITE-NE:1-1-#-#-10-1-4,1-1-#-#-34-3-7:123456::UNIDIR-
      DROP:htmr=3000,rme=DISABLE;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

A unidirectional add OUC between working tributary 2 of a 10G SONET EDGE port in slot 10 and protection tributary 3 of 2G5 SONET EDGE port 3 in slot 34 will be established by the following command: (note that for a UNIDIR-ADD OUC no path protection group from the EDGE tribs towards the INNI trib(s) is needed and thus none will be created. A UNIDIR-ADD OUC just consists of one-way point-to-point cross-connections and possible path-protected cross-connections from the INNI trib(s) to the EDGE tribs)

```
ENT-OUC:UNITE-NE:1-1-#-#-10-1-2,1-1-#-#-34-3-3:123456::UNIDIR-ADD;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Tributaries must exist

OUC command processing shall verify the tributaries specified in the command do exist.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Tributaries must belong to ports of *portClass* EDGE

OUC command processing shall verify, that both affected tributaries belong to ports of *portClass* EDGE.

Error code:

`SNVS – Status, Not in Valid State, Tributaries must be on ports of portClass EDGE`

Validation rule: Tributaries must belong to ports of equal interface standards

OUC command processing shall deny requests to establish OUCs where the ports of the tributaries are not provisioned for the same standard, either SONET or SDH – this is due to different switching criteria.

Error code:

`SNVS – Status, Not in Valid State, Different interface standards for working and protection tributaries`

Validation rule: Tributaries must not belong to ports of tributary mode ADAPTIVE

OUC command processing shall deny requests to establish OUCs where one or both ports of the tributaries are provisioned for the tributary mode ADAPTIVE.

Error code:

`SNVS – Status, Not in Valid State, port(s) for OUC in adaptive mode`

Validation rule: Tributaries must not be VCG tributaries

OUC command processing shall verify, that both affected tributaries are no VCG tributaries.

Error code:

```
IDNC – Input, Data Not Consistent, VCG-trib as input-trib in
pathprotected crossconnection
```

Validation rule: SNC Type must comply with interface standard of affected EDGE ports

OUC command processing shall deny requests to establish OUCs when a value for parameter SNC Type is given, that does not comply with the interface standard of the affected EDGE ports.

Error code:

```
SNVS – Status, Not in Valid State, Parameter value for SNC Type not
compliant to interface standard of working and protection tributaries
```

Validation rule: NN connection of matching directionality must exist on working tributary

OUC command processing shall deny requests to establish OUCs, if on the working tributary given no NN connection exists, which is of the same directionality as the OUC requested.

Error code:

```
SNVS – Status, Not in Valid State, No matching NN cross-connection
existing on working tributary
```

This means, that on a bidirectional NN connection only a bidirectional OUC can be established, while it would be possible to establish two different unidirectional OUCs on two unidirectional NN connections using the same tributary.

Validation rule: No interfering cross-connection or OUC must exist on protection tributary

OUC command processing shall deny requests to establish OUCs, if on the protection tributary given any cross-connection and/or OUC already exists interfering with the directionality of the OUC requested.

Error code:

```
SNVS – Status, Not in Valid State, Interfering cross-connection
existing on protection tributary
```

This means, that the protection tributary must not be blocked by any cross-connection at all if a bidirectional OUC is requested, and that it must not be blocked by a unidirectional add (drop) cross-connection, if a unidirectional add (drop) OUC is requested. A unidirectional add (drop) OUC would be allowed, if e.g. the tributary is blocked only by a unidirectional drop (add) cross-connection.

To be blocked by a cross-connection means for a tributary, that there must not be a cross-connection existing directly on that trib, that the trib must not be part of a cross-connection of higher bandwidth and that no ″part″ of the trib must be blocked by a cross-connection of smaller bandwidth.

Normally a protection tributary being blocked by an OUC is obviously also blocked by the cross-connections associated to this OUC. During creation of an OUC, however, there is a certain period of time, where the OUC alredy exists while the associated cross-connections do not exist yet. Also in these cases this validation will be applied, i.e. in case of creating two OUCs allocating the same protection tributary quick order, the creation of the second OUC will be rejected with this validation.

Validation rule: No OUC must already exist on working tributary to different protection tributary

OUC command processing shall deny requests to establish OUCs, if on the working tributary an OUC already exists to a different protection tributary.

Error code:

`SNVS – Status, Not in Valid State, OUC to different protection tributary already exists`

This means, that it is not possible to e.g. establish a unidirectional add OUC between a working and a protection tributary and a unidirectional drop OUC between the same working and a different protection tributary.

Validation rule: No conflicting OUC must already exist between working and protection tributary

OUC command processing shall deny requests to establish OUCs, if between the working and the protection tributary an OUC already exists that cannot coexist with the OUC requested.

Error code:

`SNVS – Status, Not in Valid State, Conflicting OUC already exists`

This covers the following cases:

- No additional OUC can be established, if a BIDIR OUC is already existing between both tribs.
- No BIDIR OUC can be established, if a UNIDIR-ADD or UNIDIR-DROP OUC is already existing between both tribs.
- No UNIDIR-ADD or UNIDIR-DROP OUC can be established, if an OUC of equal directionality is already existing between both tribs.

Due to internal system architecture there may be a state of an OUC, where it is still existing for some time after it has been deleted, before the deletion is really completed. In such a state the OUC will still be reported after RTRV-OUC. While such an OUC is existing, commands to create a new OUC on the affected working tributary or to again delete this OUC may be rejected inconsistently with error codes defined. No special validations and error codes are defined for this case.

Validation rule: Affected EDGE ports must not be in a line or equipment protection scheme

OUC command processing shall deny requests to establish OUCs, if one or both of the EDGE ports affected is involved in a line protection scheme like MSP or BLSR/MS-SPRing or an equipment protection scheme.

Error code:

`SNVS – Status, Not in Valid State, EDGE ports must not be in line or equipment protection schemes`

Validation rule: Affected EDGE port the EDGE-prt tributary is located on must not be in LOCKOUT state

OUC command processing shall deny requests to establish OUCs, if the EDGE port the EDGE-prt tributary is located on is in LOCKOUT state.

Error code:

`SNVS – Status, Not in Valid State, EDGE port carrying EDGE-prt must not be in LOCKOUT`

Validation rule: No path protection group parameters must be given for Setup of a unidirectional-add OUC

OUC command processing shall deny requests to establish unidirectional-add OUCs, if any path protection group parameter is given.

Error code:

`IDNV – Input, Data Not Valid, No path protection group parameters allowed for directionality UNIDIR-ADD`

Validation rule: No forced XC loopback must exist on EDGE-wrk tributary

OUC command processing shall deny requests to establish an OUC if, on the EDGE-wrk tributary, a forced cross-connection loopback exists.

Error code:

`SNVS – Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary`

Validation rule: No forced XC loopback must exist on EDGE-prt tributary

OUC command processing shall deny requests to establish an OUC if, on the EDGE-prt tributary, a forced cross-connection loopback exists.

Error code:

`SNVS – Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary`

ENT-OUC notification

The ENT-OUC notification will notify only the creation of the OUC itself. Cross-connections and path protection groups that are created implicitly via an OUC creation is notified via corresponding ENT-CRS-modifier and ENT-PROTN-GRP notifications.

Output format

Syntax:

ENT-OUC:*wrk_aid,prt_aid:ouc_dir,ouc_rate;*

Output parameters

Command parameters

1. **wrk_aid** – Working Tributary Access Identifier
   This parameter specifies the working tributary of the OUC.
   Type: STS1NumAID
   Range: n.a. (*all* values shall never occur as part of an output parameter)
2. **prt_aid** – Protection Tributary Access Identifier
   This parameter specifies the protection tributary of the OUC.
   Type: STSNumAID
   Range: n.a. (*all* values shall never occur as part of an output parameter)
3. **ouc_dir** – OUC Directionality
   Directionality of the OUC
   Type: Alphanumeric value set.
   Range:
   - BIDIR (default)
   - UNIDIR-ADD
   - UNIDIR-DROP
4. **ouc_rate** – OUC Rate
   The parameter *ouc_rate* indicates the rate of the OUC, which is determined by the cross-connection rate of the associated cross-connections. For a detailed description please refer to parameter rate in RTRV-CRS.
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1

- STS3
- STS12
- STS48
- STS192

Example input/output

```
   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:ENT-OUC:1-1-#-#-1-1-1,
   1-1-#-#-2-1-1:BIDIR,STS12”
;
```

```
   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:ENT-OUC:1-1-#-#-1-1-1,
   1-1-#-#-2-1-1:UNIDIR-ADD,VC4”
;
```

```
   LT-UNITE-123 00-01-01 08:00:50
A  815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:ENT-OUC:1-1-#-#-1-1-1,\
   1-1-#-#-2-1-1:UNIDIR-DROP,VC3”
;
```

ENT-PATH: Create ONNS path connection

Overview

Create an ONNS path connection with specified connection bandwidth between two client devices connected to the optical network using ONNS ports on ONNS controlled Host NE nodes with a specified protection class.

An ONNS connection may include multiple paths as in the case of a Network 1+1 connection that contains an active path and a standby path, each with its own pathID. An ONNS path is a path that connects two client EDGE ports on specified tribs through a series of NE nodes from a source node to a destination node to provide a communication service.

It is also possible to create multiple ONNS path connections of equal bandwidth, characteristics and constraints with one command, if all connections of such a group start at VCGTRIBs of one and the same VCG port and also end at VCGTRIBs of one single second VCG port.

It is also possible to create an M:N protected NN connection group with one command leading to the setup of numbers of working and protecting connections of equal bandwidth, characteristics and constraints according to a specified protection ratio M:N given in the command.

It is also possible to add a connection to an existing M:N protected NN connection group. The requested working connection has the same bandwidth, characteristics and constraints as the existing connection of the M:N protected connection group. A protection connection is setup according to the protection ratio M:N given by the selective existing M:N protected connection group.

The creation of ONNS connection via ENT-PATH is not allowed for source and/or destination ports of portclass UNI.

ENT-PATH-AUTO: Create auto-reroute ONNS connection. The **`ENT-PATH-AUTO`** command is used to establish an auto-reroute transmission connection between two Host NE EDGE ports. The **`ENT-PATH-AUTO`** command can be used to setup groups of auto-reroute protected connections.

ENT-PATH-N1PLUS1: Create 1+1 protected ONNS connection. The command **`ENT-PATH-N1PLUS1`** is used to establish a 1+1 protected transmission connection between two Host NE EDGE ports. The command **`ENT-PATH-N1PLUS1`** can be used to setup groups of 1+1 protected connections.

ENT-PATH-P1PLUS1NR: Create permanent 1+1 protected ONNS connection with non-revertive path restoration. The command **`ENT-PATH-P1PLUS1NR`** is used to establish a permanent 1+1 protected transmission connection with non-revertive path restoration (applied in case the worker or protection path fails) between two Host NE EDGE ports. The command **ENT-PATH-P1PLUS1NR** can be used to setup groups of permanent 1+1 protected connections with non-revertive path restoration.

ENT-PATH-UNPROT: Create unprotected ONNS connection. The **ENT-PATH-UNPROT** command is used to establish an unprotected transmission connection between two Host NE EDGE ports. The **ENT-PATH-UNPROT** command can be used to setup groups of unprotected connections.

ENT-PATH-YCONN: Create ONNS Y-connection. It is not possible to setup groups of Y-protected connections using the **ENT-PATH-YCONN** command. This is due to the fact, that Y-protection between VCGs does not make sense. The **ENT-PATH-YCONN** command is used to establish a Y-protected transmission connection between three Host NE EDGE ports.

ENT-PATH-MBYN: Create an M:N protected connection group of directional connections. The **ENT-PATH-MBYN** command is used to establish a number of working and protecting directional connections to form an M:N protected connection group.

ENT-PATH-MBYN: Add a connection to an existing M:N protected connection group. The **ENT-PATH-MBYN** command is used to add a working and a protecting connection (if the protection ratio demands this) to an existing M:N protected connection group. The existing M:N protected connection group is identified by its group identifier (nodeID&groupIndex).

The **ENT-PATH-protClass** command is used for the setup of adaptive ONNS connections. For the setup of adaptive ONNS connections, each parameter of the **ENT-PATH-protClass** command is applied for the adaptive ONNS connection as a whole and not for any separate constituent signal. Corresponding restrictions can be found in "Error responses" (p. 3-1739).

Properties

Login privilege: User Privilege Code: (UCFC/UCAL): P3

Abortable: No

Input format

Syntax of ENT-PATH-UNPROT

The **ENT-PATH-UNPROT** command is used to establish unprotected connections:

Command syntax: **ENT-PATH-UNPROT:***srcTid:[srcAid]:ctag::[srvID]:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:destPortID:[nodeListUsage1,nodeList1]:[srlgExcludeList],[linkIncludeID][::specBlock]*;

Syntax of ENT-PATH-AUTO

The **ENT-PATH-AUTO** command is used to establish auto-rerouting connections:

Command syntax: **ENT-PATH-AUTO:***srcTid:[srcAid]:ctag::[srvID]:pdir: bandwidth:maxDelay:srcNID:srcPortID:destNID:destPortID: [nodeListUsage1,nodeList1]:[srlgExcludeList] ,[linkIncludeID] [::specBlock]*;

Syntax of ENT-PATH-N1PLUS1

The command **ENT-PATH-N1PLUS1** is used to establish 1+1 protected connections:

Command syntax: **ENT-PATH-N1PLUS1:***srcTid:[srcAid]:ctag::[srvID]:pdir: bandwidth:maxDelay:srcNID: srcPortID:destNID:destPortID: [nodeListUsage1,nodeList1]: [srlgExcludeList],[linkIncludeID]: [nodeListUsage2,nodeList2] [:specBlock]*;

Syntax of ENT-PATH-P1PLUS1NR

The command **ENT-PATH-P1PLUS1NR** is used to establish permanent 1+1 protected connections with non-revertive path restoration:

Command syntax:
ENT-PATH-P1PLUS1NR:*srcTid:[srcAid:ctag]::[srvID]:pdir:bandwidth:maxDelay:srcNID: srcPortID:destNID:destPortID:[nodeListUsage1,nodeList1]: [srlgExcludeList],[linkIncludeID]:[nodeListUsage2,nodeList2][:specBlock];*

Syntax of ENT-PATH-YCONN

The **ENT-PATH-YCONN** command is used to establish bridged Y-connections:

Command syntax: **ENT-PATH-YCONN:***srcTid:[srcAid]:ctag::[srvID]:pdir: bandwidth:maxDelay:srcNID: srcPortID:destNID,brDestNID: destPortID,brDestPortID:[nodeListUsage1,nodeList1]: [srlgExcludeList],[linkIncludeID]:[nodeListUsage2,nodeList2] [:specBlock]*;

Syntax of ENT-PATH-MBYN

The **ENT-PATH-MBYN** command is used to establish an M:N protected NN connection group or to add a worker connection to an existing M:N protection group:

Command syntax: **ENT-PATH-MBYN:**
srcTid:[srcAid]:ctag::[srvID]:[connID]:pdir:bandwidth:maxDelay: srcNID: srcPortID[,srcPortID1,...,srcPortID63]:destNID:destPortID[,destPortID1,..., destPortID63]:[nodeListUsage1,nodeList1]: [srlgExcludeList],[linkIncludeID]::specBlock;

REPT PTHCHG is generated

The **ENT-PATH** command shall generate a REPT-PTHCHG message.

REPT PTHCHG for groups of connections

The **ENT-PATH** command for groups of connections shall generate one separate REPT-PTHCHG message for each connection successfully setup.

Description

The ENT-PATH-*protClass* command is used to establish one or more optical transmission path(s) between two valid client routers or traditional nodes in the network that are attached to Host NE EDGE ports.

ENT-PATH creates an optical connection with specified connection bandwidth between two client devices connected to the optical network using ONNS ports / tribs on ONNS controlled Host NE nodes.

The *protClass* of the **ENT-PATH** command indicates what type of protection should be provided with the light path(s) to be established.

The path setup commands are:

- ENT-PATH-UNPROT
- ENT-PATH-N1PLUS1
- ENT-PATH-P1PLUS1
- ENT-PATH-YCONN
- ENT-PATH-AUTO
- ENT-PATH-MBYN

ENT-PATH-UNPROT sets up an unprotected path from source *srcNID*, *srcPortID* to destination *destNID*, *destPortID*. This is a basic ONNS connection with no ONNS protection response to failure except a REPT-PTHEVT. ENT-PATH-UNPROT shall set up a group of unprotected connections from srcNID, srcPortID to destination destNID, destPortID, if srcPortID specifies a list of VCG tributaries of one and the same EDGE VCG in *srcNID* and *destPortID* specifies an EDGE VCG in *destNID*.

For the setup of a group of unprotected connections with ENT-PATH-UNPROT the parameter grpSetupType can be used to specify, whether the total cost of all connections of the grouped setup or the number of routes taken by the connections of the grouped setup is minimized. This parameter is ignored, if the grouped setup actually only consists of the setup of one single unprotected connection, as in this case both alternatives will lead to equal results. This parameter shall also be ignored, if explicit routing is requested by setting parameter nodeListUsage1 to EXPLICIT, as also in this case both alternatives would make no difference.

ENT-PATH-AUTO sets up an auto-rerouting nonrevertive protected path.

ENT-PATH-AUTO, non-revertive: This is an ONNS nonrevertive auto-rerouting connection that ONNS will restore using ONNS mesh restoration techniques. The failed path will be bridged with a new path that satisfies all constraints applied to the connection. The new connection will be verified that it meets the service level agreements (SLA), and the failed path will be torn down. Link resources of the failed path will be returned to the available resource pool. ENT-PATH-AUTO, nonrevertive, shall set up a group of nonrevertive restoration protected connections from *srcNID*, *srcPortID* to destination *destNID*, *destPortID*, if *srcPortID* specifies a list of VCG tributaries of one and the same EDGE VCG in srcNID and *destPortID* specifies an EDGE VCG in *destNID*.

ENT-PATH-AUTO, revertive: This is an ONNS revertive auto-rerouting connection that ONNS will restore using ONNS mesh restoration techniques. The failed path will be bridged with a new path that satisfies all constraints applied to the connection. The new connection will be verified that it meets the service level agreements (SLA), and the resources of the original path will be kept. When the original path becomes good again, reversion takes place manually or automatically depending on the *switchback* parameter setting. The restoration path will be down after successful reversion. ENT-PATH-AUTO, revertive, shall set up a group of revertive restoration protected connections from *srcNID*, *srcPortID* to destination *destNID*, *destPortID*, if *srcPortID* specifies a list of VCG tributaries of one and the same EDGE VCG in *srcNID* and *destPortID* specifies an EDGE VCG in *destNID*.

ENT-PATH-N1PLUS1 sets up both paths of a network 1+1 protected connection. ONNS sets up two node disjoint and (if applicable) SRLG-diverse paths through the network that share a common ingress(/egress) port (/connPoint) on the source NE node and a common egress (/ingress) port (/connPoint) on the destination NE node. The N1PLUS1 setup shall support the setting of the path protection group parameters:

- revertive / non-revertive mode
- wait to restore timer
- hold off timer.

These parameters will be set consistently at both ends of the 1+1 connection in case of bidirectionality. The setup of the path protection group parameters via ENT-PATH-N1PLUS1 simplifies the provisioning of these parameters for ONNS 1+1 protection connections. These parameters are part of the path protection group and still provisionable via ED-PROTN-GRP. These parameters are not administrated by ONNS and for this reason are not specified as TL1 output parameters.

ENT-PATH-N1PLUS1 shall set up a group of 1+1 protected connections from *srcNID*, *srcPortID* to destination *destNID*, *destPortID*, if *srcPortID* specifies a list of VCG tributaries of one and the same EDGE VCG in *srcNID* and *destPortID* specifies an EDGE VCG in *destNID*.

ENT-PATH-P1PLUS1 shall set up both paths of a permanent 1+1 protected path. ONNS sets up two node disjoint and (if applicable) SRLG-diverse paths through the network that share a common ingress(/egress) port (/connPoint) on the source NE node and a common egress (/ingress) port (/connPoint) on the destination NE node. In case that the worker or protection path of this 1+1 protection connection fails a new non-revertive autoreroute path will be setup.

ENT-PATH-P1PLUS1 shall set up a group of permanent 1+1 protected connections from *srcNID*, *srcPortID* to destination *destNID*, *destPortID*, if *srcPortID* specifies a list of VCG tributaries of one and the same EDGE VCG in *srcNID* and *destPortID* specifies an EDGE VCG in *destNID*.

ENT-PATH-YCONN sets up both paths of a bridged Y-connection from one source NE node to two different destination NE nodes. ONNS sets up two node disjoint and (if applicable) SRLG-diverse paths through the network that share a common ingress(/egress) port (/connPoint) on the source NE node, but terminating at two different destination NE nodes with its own egress(/ingress) port (/connPoint).

ENT-PATH-MBYN shall set up a group of connections consisting of a number of working connections and a number of protecting connections from source *srcNID* with a list of tributaries to destination destNID with a list of tributaries.

The list of tributaries are given as:

- a list of *srcPortIDs* with a single, a list or a range of tributaries per *srcPortID* for the source
- one *srcPortID* with a list or a range of tributaries for the source
- a list of *destPortIDs* with a single, a list or a range of tributaries per *destPortID* for the destination
- one *destPortID* with a list or a range of tributaries for the destination.

ENT-PATH-MBYN shall add one working connection to an existing M:N protected connection group identified by its connection group identity (nodeIDgroupIndex) from source *srcNID*, *srcPortID* to destination *destNID*, *destPortID*. The requested working connection shall have the same bandwidth, characteristics and constraints as the existing connections of the M:N protected connection group. A protection connection is setup according to the protection ratio M:N given by the existing M:N protected connection group if necessary.

User Defined Connection Alias: A user defined connection alias (an alternate name) may be optionally specified with the **`ENT-PATH`** command. The alias is specified in the spec block via the parameter *connAlias*. If *connAlias* is not specified, the system will treat it as NULL. If the specified *connAlias* already exists in the source node, the command will be denied. *connAlias* is also returned as an output parameter. If none was defined a NULL is returned.

User Defined Connection Alias when setting up VCG connection groups: If a user defined connection alias (an alternate name) is specified with an ENT-PATH command setting up a group of connections, this alias is inherited to the connections of the group in the following way for VCG connection groups: Each connection shall get as connection alias the alias given in the command appended by "_" (underscore) and the tributary number of the corresponding tributary in the A-node (consisting of 3 digits), i.e. 4 characters are appended to the connection alias string. If the string resulting from appending the 4 characters to the alias string given in the command would exceed the total length allowed for a connection alias, the original string given in the command will first be truncated at its tail by such a number of characters, so that the length of the resulting connection alias (i.e. appended by 4 characters) will have the maximum length allowed. If such a connection alias will not be unique any more in the node, the setup of the corresponding connection is denied.

Example:

The following connection alias is supposed to have maximum length allowed: "Thisisalong connection alias…thisisalongconnectionalias". If given as alias in a group setup command, the aliases for the connections of this group would then look as follows:

"Thisisalong connection alias…thisisalongconnectional_001"

"Thisisalong connection alias…thisisalongconnectional_002"

…

"Thisisalong connection alias…thisisalongconnectional_190"

User Defined Connection Alias when setting up M:N connection groups: If a user defined connection alias (an alternate name) is specified with an ENT-PATH command setting up a group of connections, this alias is inherited to the connections of the group in the following way for M:N protected connection groups: Each working connection shall get as connection alias the alias given in the command appended by "_wrk_" and an index unique within the working connections in the group (consisting of 3 digits with leading zeroes), i.e.8 characters are appended to the connection alias string. Each protecting connection shall get as connection alias the alias given in the command appended by "_prt_" and an index unique within the protecting connections in the group (consisting of 3 digits with leading zeroes), i.e. 8 characters are appended to the connection alias string. If the string resulting from appending the 6 to 8 characters to the alias string given in the command would exceed the total length allowed for a connection alias, the original string given in the command will first be truncated at its tail by such a number of characters, so that the length of the resulting connection alias (i.e. appended by 8 characters) will have the maximum length allowed. If such a connection alias will not be unique any more in the node, the setup of the corresponding connection group is denied.

User Defined Connection Alias when adding M:N connections: If a user defined connection alias (an alternate name) is specified for a connection extension of a M:N protection group, this alias is inherited to the added connection of the group in the following way: The added working connection shall get as connection alias the user defined alias appended by ″_wrk_″ and a free index unique within the working connections with the same alias (consisting of 3 digits with leading zeroes), i.e. 8 characters are appended to the connection alias string.

A possible new protecting connection shall get as connection alias the user defined alias appended by ″_prt_″ and a free index unique within the protecting connections with the same alias (consisting of 3 digits with leading zeroes), i.e. 8 characters are appended to the connection alias string. If the string resulting from appending the 6 to 8 characters to the alias string given in the command would exceed the total length allowed for a connection alias, the original string given in the command will first be truncated at its tail by such a number of characters, so that the length of the resulting connection alias (i.e. appended by 8 characters) will have the maximum length allowed.

The creation of ONNS connection via ENT-PATH is not be allowed for source and/or destination ports of *portclass* UNI.

Input parameters

Command parameters

1. **`protClass`** – (modifier of ENT-PATH)
   Path protection class defines the type of restoration protection assigned to the path connection during path setup.
   UNPROT: path from source to destination is un-protected.
   AUTO: path from source to destination is auto-reroute path connection.
   N1PLUS1: paths from source to destination is set up as a network 1+1 connection.
   YCONN: paths from the same source to two different destinations are set up as a bridged connection.
   MBYN: group of M:N protected working and protecting paths from the same source node to the same destination node.
   P1PLUS1NR: paths from source to destination is set up as a permanent 1+1 protected connection with an additional non-revertive restoration path in case the worker or protection path fails.
2. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
3. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.

4. **ctag** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing “.”).
5. **srvID** – Service Identifier
   This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.
6. **connID** – Connection ID
   See definition of the M:N protection group index of the input parameter *connID* in RTRV-PATH. Only relevant for MBYN.
   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
7. **pdir** – Path Direction Topology
   Unidirectional A-Z (1WAY) or Bidirectional (2WAY).
   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default), 2WAY
8. **bandwidth** – Requested bandwidth / signalRate
   ONNS bandwidth is equal to or greater than the minimum cross connect rate (minXCrate) of the port type. Note: Only SONET switching levels in enumeration. For actual cross-connections ONNS adapts to SDH formats automatically if the ingress port is SDH.
   Type: Value set.
   Values: STS1, STS3, STS12, STS48, STS192.
9. **maxDelay** – Maximum transport delay
   Maximum transport delay through all nodes in the path. Total path delay must not exceed this value.
   Range: 0 (default), is used to select no delay limit; >0, maximum transport delay in milliseconds.
10. **srcNID** – Source nodeID
   ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
   Range: Four dot separated numeric strings. 0 – 255 for each numeric string.

11. **`srcPortID`** – Source PortID
ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
Value: dash-separated list of 5 numeric strings
Example: 1-1-3-1-4
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range, if port # specifies a VCG: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # List (Any sequence of Tributary # or Tributary # Range sep. by & and in ascending order), Tributary # Range (Tributary#1&&Tributary#2), Tributary # (1 – 192).
Example: 1-1-15-6-1&3&5&&7&9&13&&18&21. The port AID 1-1-15-6 must denote a VCG.
Range, if protClass is equal ″MBYN″: Bay/Frame #(1), Shelf #(1), Slot/Pack #(1..39), Port #(1..16), Tributary # List (Any sequence of Tributary # or Tributary # Range sep. by &). Tributary # Range (Tributary#1&&Tributary#2). Tributary #(1..768).
Example: 1-1-3-1-1&2&8&13&&18&6&20&3&&5. The port AID 1-1-3-1 must not be a VCG In this example the bandwidth is supposed to be VC3/STS1
If *protClass* is equal ″MBYN″ the *srcPortID* can be a list up to 64 *portIDs*. If *protClass* is equal ″MBYN″ and a valid M:N *groupID* in the *connID* parameter was given only one *srcPortID* with one tributary is allowed.

12. **`destNID`** – Destination nodeID
ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

13. **`destPortID`** – Destination PortID
ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range, if Port # specifies a VCG: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # (ALL).
The ALL qualifier for Tributary # in *destPortID* must be given, when a Tributary # List with more than 1 entry is given in *srcPortID*.
Range, if *protClass* is equal ″MBYN″: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1..39), Port # (1..16), Tributary # List (any sequence of Tributary # or Tributary # Range separated by &), Tributary # Range (Tributary#1&&Tributary#2), Tributary # (1..768).

Example: -1-3-1-1&2&8&13&&18&6&20&3&&5. The port AID 1-1-3-1 must not be a VCG. In this example the bandwidth is supposed to be VC3/STS1.
If *protClass* is equal ″MBYN″ the *destPortID* can be a list up to 64 portIDs. If *protClass* is equal ″MBYN″ and a valid M:N *groupID* in the *connID* parameter was given only one destPortID with one tributary is allowed.

14. **`brDestNID`** – Bridged Destination NodeID
ONNS bridged destination nodeID. Together with the following brDestPortID field defines the second associated Egress connPoint that connects the destination node to the edge/gateway switch or client router in case of protectionClass YCONN.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

15. **`brDestPortID`** – Bridged Destination PortID
ONNS bridged destination node egress *portID* in case *protectionClass* = YCONN. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the brDestNID egress connPoint. 4.1; 5; 6
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).

16. **`nodeListUsage1`** – Node List Usage 1
For connection setup NN used/uses the list of nodes included in the *nodeList1* array block in a manner specified by the *nodeListUsage1* variable. Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command.
An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent *nodeID* will be ignored in EXCLUDE *nodeList1* during path setup. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
Range:
    - EXPLICIT, *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.)
    - EXCLUDE, nodes in the *nodeList1* must all be excluded from the path created. In the case when the *protectionClass* is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the *nodeList1* below also apply to the second or restoration path.

17. **`nodeList1`** – Node List 1
When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidx]).
When *nodeListUsage1* = EXCLUDE: List of nodes that must all be excluded from both paths. The *nodeID* values are shown in the from: nid1([&nid2& …&nidx]).
If the preceding required *nodeListUsage1* field is not supplied, this field must not be present. If user specifies empty EXCLUDE *nodeList1*, the connection should use indirect path from source node to destination node (i.e., at least one intermediate

node is needed). If user specifies empty EXPLICIT *nodeList1*, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
Range: nid1([&nidx])* or empty list (where X <= 25 when *nodeListUsage1* = EXPLICIT) (where X <= 25 when *nodeListUsage1* = EXCLUSIVE)

18. **`srlgExcludeList`** – SRLG Exclude List
List of Shared Risk Link Group identifiers that are to be used to exclude links from the connection path(s) being created.
Type: &-separated list of numeric strings | empty | 0
Example: 33&49&64
Range: up to 10 numeric strings, each in the range 1 – 65535. The value 0 (default) or an empty field indicates no constraint are applied.

19. **`linkIncludeID`** – VPN Identifier
A connection constraint link property such as *linkIncludeID* that must be valid to include the link in the connection path(s) to be created. *linkIncludeID* is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
Range: 1 – 65535. 0 (default) or an empty field indicates no VPN constraint is applied.
Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.
This is an optional input parameter in the **`ENT-PATH`** command.

20. **`nodeList Usage2`** – Node List Usage 2
For connection setup NN used/uses the list of nodes included in the *nodeList2* array block in a manner specified by the *nodeListUsage2* variable. Both, *nodeListUsage2* and *nodeList2*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
This parameter shall also be allowed when protectionClass is P1PLUS1NR.
Range: EXPLICIT, *nodeList2* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.).

21. **`nodeList2`** – Node List 2
When nodeListUsage2=EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The nodeID values are shown in the form: nid1([[&nid2& …&nidx]]).
If the preceding required *nodeListUsage2* field is not supplied, this field must not be present. If user specifies empty EXPLICIT *nodeList2*, the connection should use a direct path from source to destination node (i.e., no intermediate node is

allowed).
This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
This parameter shall also be allowed when protectionClass is P1PLUS1NR.
Type: &-separated list of *nodeIDs*. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
Range: nid1([&nidX])* or empty list where X <= 8 when *nodeListUsage1* = EXPLICIT.

Spec block parameters

1. **`connMode`** – Connection Mode
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the auto-reroute or 1+1 protected connection.
   This parameter can be used with the ENT-PATH-AUTO and ENT-PATH-N1PLUS1 command. This parameter can be used with the ENT-PATH-P1PLUS1NR command to specify, whether revertive or non-revertive mode is used as operation mode of the 1+1 protection of the permanent 1+1 protected connection.
   Type: Alphanumeric value set.

   Range:

   - NONREVERTIVE (default)
   - REVERTIVE.
2. **`rstrPriority`** – Restoration Priority
   This parameter describes the restoration priority of the connection.
   Range: 0 (default) – 7. Lower values represent higher priority.
3. **`connAlias`** – connection alias
   This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83.
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes.
4. **`disjointness`**
   The disjointness determines the path calculation logic.
   Type: Alphanumeric value set.

   Range:

   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.

   Maximal SRLG disjoint path calculation is not supported by the current path calculation algorithms.
   Impact on disjointness parameter values when protClass is set to N1PLUS1 or YCONN
   When *protClass* is set to N1PLUS1 or YCONN, only the *disjointness* parameter

values COMPLETE and FULLYSRLG is allowed.

Impact on disjointness parameter values when protClass is set to P1PLUS1NR

When *protClass* is set to P1PLUS1NR, only the *disjointness* parameter values COMPLETE and FULLYSRLG is allowed.

Default:

- COMPLETE: for protectionClass N1PLUS1 and YCONN.
- MAXIMAL: for protectionClass AUTONR and AUTORV.

The parameter is only applicable when *protectionClass* != UNPROT.
Default: COMPLETE: for *protectionClass* P1PLUS1NR

5. **`grpSetupType`** – grouped Setup Type
The grouped Setup Type specifies, whether for all NN connections of a grouped setup the total cost of all connections or the number of routes used by the connections are minimized.
Both alternatives will lead to equal results in the following cases:
   - Only one connection is setup with a grouped setup
   - There is enough bandwidth available via the cheapest route for all connections of a grouped setup.

   Type: Alphanumeric value set.
Range: MINCOST (default): the total cost for all connections of the grouped setup is minimized. MINROUTES: the number of routes used by the connections of the grouped setup is minimized.
6. **`protRatio`** – Protection ratio
For each working or protecting connection of an M:N protected connection group this parameter will report the concrete protection ratio as specified on creation of the group. For connections not being member of an M:N protected connection group, this parameter is not reported at all.
Type: Alphanumeric value set
Range. BY1 (ratio of 1 to 1 in number of protecting and working connections), 1BY2 (ratio of 1 to 2 in number of protecting and working connections), 1BY3 (ratio of 1 to 3 in number of protecting and working connections).
7. **`wtr`** – Wait to Restore Timer
This parameter specifies the value of the wait-to-restore time for revertive 1+1 protection switching. Parameter applies to 1+1 protected and permanent 1+1 protected connections.
The parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
Type: numeric string [minute]
Range: 0 – 60.
The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
Default: 0

8. **`htmr`** – Hold Off Timer
   This parameter specifies the value of the hold-off timer for 1+1 protection switching. Parameter applies to 1+1 protected and permanent 1+1 protected connections.
   The parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms]
   Range for high order path protection groups:
   - 0 -99 ms in 1 ms steps
   - 100 -10000 ms in 100 ms steps

   Range for low order path protection groups:
   - 0, 25, 50, 75 ms (thus in steps of 25 ms)
   - 100 -10000 ms in 100 ms steps.

   Default: None.
   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block. Default value: 0.
9. **`switchBack`** – revertive restoration path switch back
   This parameter describes how a revertive restoration path is switched back to the original path.
   Type: Alphanumeric value set.
   Range:
   - AUTO: path is automatically switched back after wait to revert time.
   - MAN: path has to be manually switched back triggered by the TL1 command **`ED-PATH-ROLL`**.

   Default: AUTO.
   This parameter is only allowed for the **`ENT-PATH-AUTO`** command with *connMode* set to revertive.

Output format

System response format to ENT-PATH-protClass on grouped setup of VCG connections

For the ENT-PATH request for setting up a group of VCG connections the response is as follows:

- If the command as a whole is denied due to e.g. syntax or input failure, the command is denied simply giving completion code DENY together with the error reason
- If setting up of all connection of the group succeded, the command is completed giving completion code COMPLD and all connections of the group is listed. For each connection the parameters are given as would be given in a reply to a successful ENT-PATH-protClass command for a single connection.

- If not all connections could be set up successfully, but the setup of a number of connections failed, the command is completed giving completion code PRTL and giving a list of the following informations: For each connection whose setup succeeded the parameters are given as would be given in a reply to a successful ENT-PATH-protClass command for a single connection. For each connection whose setup failed the tributary ID is given together with the error reason for the failure as would be given in a reply to a failed ENT-PATH-protClass command for a single connection.
- If the setup of all connections failed individually, the command is denied giving completion code DENY and all connections of the group is listed. For each connection the tributary ID is given together with the error reason for the failure as would be given in a reply to a failed ENT-PATH-protClass command for a single connection.

System response format to ENT-PATH MBYN

The ENT-PATH-MBYN request for setting up an M:N protected group of connections normally will result in the setup of multiple working and protecting connections.

The following system response behavior is valid:

- If the command as a whole is denied due to e.g. syntax or input failure, the command is denied simply giving completion code DENY together with the error reason
- If it is not possible to setup all required connections but only a part of them, the whole group cannot be created and thus connections of the group that have already been setup must be torn down again (roll-back). In this case the command is denied giving completion code DENY together with one entry for each connection the setup of which failed together with the error reason. Additionally the reply shall contain one entry for each connection that had been setup successfully but had to be torn down for roll-back reason indicated by the specific error reason E061. Each working connection in such a reply is identified by the corresponding EDGE tributary AID as ″tribAID=<tributaryAID>″. Each protecting connection in such a reply is identified using the index of its connection ID by ″tribAID=#&<index>″
- If setting up of all connections of the group succeded, the command is completed giving completion code COMPLD and all connections of the group is listed. For each connection the parameters are given as would be given in a reply to a successful ENT-PATH-UNPROT command for a single connection.

System response format to ENT-PATH-MBYN for a added connection

The ENT-PATH-MBYN request for setting up a additional connection to a M:N protected group of connections will result in the setup of a working connection and possibly in the setup of a protecting connection. The following system response behavior is valid:

- If the command as a whole is denied due to e.g. syntax or input failure, the command is denied simply giving completion code DENY together with the error reason
- If it is not possible to setup possible protection connection or the working connection, the already setup connection must be torn down again (roll-back). In this case the command is denied giving completion code DENY together with an entry for the connection of which failed together with the error reason. Additionally the reply shall contain one entry for the connection that had been setup successfully but had to be torn down for roll-back reason indicated by the specific error reason E061.
  The working connection in such a reply is identified by the corresponding EDGE tributary AID as ″tribAID=<tributary AID>″. The protecting connection in such a reply is identified using the index of its connection ID by ″tribAID=#&<index>″
- If setting up of the added connections to the group together with a possible setup of a protection connection succeded, the command is completed giving completion code COMPLD and both additional connections of the group is listed. For both connection the parameters are given as would be given in a reply to a successful ENT-PATH-UNPROT command for a single connection.

System response format to ENT-PATH-N1PLUS1

The ENT-PATH-N1PLUS1 shall respond with the completion code PRTL if the path protection group parameters (revertive mode, wait to restore timer or hold off timer) setup with ENT-PATH-N1PLUS1 request are not successfully provisioned. The response shall contain the normal system response format to ENT-PATH-N1PLUS1 and additionally a failure information about the missing path protection group parameter provisioning.

Normal system response to ENT-PATH-AUTO/ENT-PATH-UNPROT

If the **ENT-PATH** request completes successfully, the following normal completion response is returned:

```
ENT-PATH-
    AUTO:srcTid:[srcAid]:ctag::srvID:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:dest

   srcTid date time
M  ctag COMPLD
   ″ConnID=connID,ConnStatus=connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,,
   SrlgExclude=srlgExcludeList,IncludeID=linkIncludeID,[SpecBlock],,
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
```

```
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1"
;
```

Note: *nodeListUsage1* = will be translated to either *Exclude* = or *Explicit*=, depending on which type of *nodeList1* was received as input. In the example the nodeList1 = Node5ID&Node61D.

Normal system response to ENT-PATH-N1PLUS1

If the **ED-PATH-N1PLUS1** request completes successfully, the following normal completion response is returned:

```
ENT-PATH-
      N1PLUS1:srcTid:[srcAid]:ctag::[srvID]:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:de
[NodeListUsage2,nodeList2]:specBlock;

   srcTid date time
M  ctag COMPLD
   "ConnID=connID,ConnStatus=connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=connAlias],

      MaxDelay=maxDelay,[SrvID=srvID],exclude=Node5ID&Node6ID,[nodeListUsage2=nodeList2],SrlgEx
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1:
   PathID=pathID2,State=pst2&sst2&tst2,
   NodeList=pNodeList2"
;
```

Note: *nodeListUsage1* = will be translated to either *Exclude* = or *Explicit*=, depending on which type of *nodeList1* was received as input.

The system response for **ENT-PATH-P1PLUS1NR** corresponds with the system response of **ENT-PATH-N1PLUS1**.

System response to ENT-PATH-N1PLUS1 with successful connection setup but unsuccessful path protection group parameter provisioning

If the **ENT-PATH** request completes partly, the following partial response is returned:

```
ENT-PATH-
      N1PLUS1:srcTid:[srcAid]:ctag::[srvID]:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:de

   srcTid date time
M  ctag PRTL
   "ConnID=connID,ConnStatus=connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=connAlias],

      MaxDelay=maxDelay,[SrvID=srvID],[nodeListUsage1=nodeList1],[nodeListUsage2=nodeList2],Srl
```

```
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID:
   PathID=pathID1,State=pst1&sst1&tst1,
   NodeList=pNodeList1:
   PathID=pathID2,State=pst2&sst2&tst2,
   NodeList=pNodeList2"

   "E150::\"Status, Path Protection Group parameters Not Provisioned\","
;
```

Note: *nodeListUsage1* = will be translated to either *Exclude* = or *Explicit*=, depending on which type of *nodeList1* was received as input.

The system response for **ENT-PATH-P1PLUS1NR** corresponds with the system response of **ENT-PATH-N1PLUS1**.

Normal system response to ENT-PATH-YCONN

If the **ENT-PATH-YCONN** request completes successfully, the following normal completion response is returned:

```
RTRV-CKT:UNITE1:1-1-#-#-22-1-1:123;
IP 123
<


   UNITE1 04-12-28 00:22:15
M  123 COMPLD
   "ring0"
   "UNITE0,1,REMAID,REM-1-1-#-#-22-1-TS1,UNIDIR,STS1:rappl=ADD"
   "UNITE1,0,1-1-#-#-22-1-1,1-1-#-#-23-1-1,UNIDIR,STS1:rappl=THRU"
   "UNITE2,2,REM-1-1-#-#-24-1-TS1,PIF
;
```

Note: *nodeListUsage1* = will be translated to either *Exclude* = or *Explicit*=, depending on which type of *nodeList1* was received as input.

System response to ENT-PATH-AUTO/ENT-PATH-UNPROT for group of connections with successful setup of all connections affected

If the **ENT-PATH** request for setting up a group of N connections completes successfully for all connections, the following normal completion response is returned.

Note: There is no restriction wrt the order of the connection information sent within the reply. If one reply would exceed the maximum length allowed for a TL1 message, whole connection records are distributed into separate messages.

```
ENT-PATH-
     AUTO:srcTid:[srcAid]:ctag::srvID:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:dest

   srcTid date time
```

```
M  ctag COMPLD
   ″ConnID=1connID,ConnStatus=1connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=1connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],SrlgExclude=srlgExcludeList,
   IncludeID=linkIncludeID,[SpecBlock],,
   SrcNID=srcNID,1srcPortID,DestNID=destNID,1destPortID:
   PathID=1pathID1,State=1pst1&sst1&tst1,
   NodeList=1pNodeList1″

   ″ConnID=2connID,ConnStatus=2connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=2connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],SrlgExclude= srlgExcludeList,
   IncludeID=linkIncludeID,[SpecBlock],,
   SrcNID=srcNID,2srcPortID,DestNID=destNID,2destPortID:
   PathID=2pathID1,State=2pst1&sst1&tst1,
   NodeList=2pNodeList1″
   .
   .
   .

   ″ConnID=NconnID,ConnStatus=NconnStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=NconnAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],SrlgExclude=srlgExcludeList,
   IncludeID=linkIncludeID,[SpecBlock],,
   SrcNID=srcNID,NsrcPortID,DestNID=destNID,NdestPortID:
   PathID=NpathID1,State=Npst1&sst1&tst1,
   NodeList=NpNodeList1″
;
```

System response to ENT-PATH-AUTO/ENT-PATH-UNPROT for group of connections with successful setup of a part of the connections affected

If the **ENT-PATH** request for setting up a group of N connections completes partially, i.e. it completes for M connections and fails for N-M connections, the following completion response is returned:

Note: There is again no restriction wrt the order of the connection information (either success or failure) sent within the reply. If one reply would exceed the maximum length allowed for one TL1 message, whole connection records are distributed into separate messages.

```
ENT-PATH-
      AUTO:srcTid:[srcAid]:ctag::srvID:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:destPor

   srcTid date time
M  ctag PRTL
```

```
   ″ConnID=1connID,ConnStatus=1connStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=1connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[SpecBlock],,
   SrcNID=srcNID,1srcPortID,DestNID=destNID,1destPortID:
   PathID=1pathID1,State=1pst1&sst1&tst1,
   NodeList=1pNodeList1″
   .
   .
   .

   ″ConnID=MconnID,ConnStatus=MconnStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=MconnAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[SpecBlock],,
   SrcNID=srcNID,MsrcPortID,DestNID=destNID,MdestPortID:
   PathID=MpathID1,State=Mpst1&sst1&tst1,
   NodeList=MpNodeList1″

   (M+1)ErrorCode, tribAID=(M+1)tribAID
   ″(M+1)msgCode:(M+1)srvID:\″ErrorText String\″,[errorData]″
   .
   .
   .

   NErrorCode, tribAID=NtribAID,
   ″NmsgCode:NsrvID:\″ErrorText String\″,[errorData]″
;
```

System response to ENT-PATH-MBYN for an M:N protected group of working and protecting connections with successful setup of all connections affected

If the **ENT-PATH-MBYN** request for setting up an M:N protected group of connections completes successfully for all connections, the following normal completion response is returned:

Note: There is no restriction wrt the order of the connection information sent within the reply. If one reply would exceed the maximum length allowed for a TL1 message, whole connection records are distributed into separate messages. In this example it is assumed that 2 working connections to be 1:2 protected are setup.

```
ENT-PATH-
      AUTO:srcTid:[srcAid]:ctag::srvID:pdir:bandwidth:maxDelay:srcNID:srcPortID:destNID:dest

   srcTid date time
M  ctag PRTL
   ″ConnID=1connID,ConnStatus=1connStatus,pdir,protectionClass,bandwidth,
```

```
   [ConnAlias=1connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[SpecBlock],,
   SrcNID=srcNID,1srcPortID,DestNID=destNID,1destPortID:
   PathID=1pathID1,State=1pst1&sst1&tst1,
   NodeList=1pNodeList1"
   .
   .
   .

   "ConnID=MconnID,ConnStatus=MconnStatus,pdir,protectionClass,bandwidth,
   [ConnAlias=MconnAlias],
   MaxDelay=maxDelay,[SrvID=srvID],Exclude=Node5ID&Node6ID,
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[SpecBlock],,
   SrcNID=srcNID,MsrcPortID,DestNID=destNID,MdestPortID:
   PathID=MpathID1,State=Mpst1&sst1&tst1,
   NodeList=MpNodeList1"

   (M+1)ErrorCode, tribAID=(M+1)tribAID
   "(M+1)msgCode:(M+1)srvID:\"ErrorText String\",[errorData]"
   .
   .
   .

   NErrorCode, tribAID=NtribAID,
   "NmsgCode:NsrvID:\"ErrorText String\",[errorData]"
;
```

For adding a working connection the optional parameter *connID* is provisioned. The reply will contain the parameters of the setup working connection and possibly of a setup protection connection.

The **ENT-PATH** command for grouped connection setup shall send a maximum of 8 connection records per block. On all large output message blocks but the last, the continuation completion code COMPLD, PRTL or DENY is used depending on the outcome for the whole command and the semicolon (;) is replaced with a (>) terminator. Then on the last record block, the completion code COMPLD, PRTL or DENY is used depending on the outcome for the whole command with the final semicolon (;) data block terminator.

Note: This **ENT-PATH** command for grouped connection setup can produce large amounts of data. This will require proper sizing of the receive buffer of the OS system.

Output parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   This parameter is set to the date when the output message is generate: YY-MM-DD where YY=2000-2099, MM=01-12, DD=01-31.
3. **`time`**
   This parameter is set to the time when the output message is generated: HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.
4. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
5. **`completionCode`**
   Standard TL1 completion code result:
   - COMPLD: for success (default)
   - PRTL: for partial success / failure, see section ERROR RESPONSES for details
   - DENY: in case of errors, see section ERROR RESPONSES for details
6. **`connID`** – Connection ID
   This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and IngressPort# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.
   Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with
   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary# 192.

- Index: Numeric string 0-4294967295.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203&1-13-15-1&271584.

Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

  Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 9), Tributary # ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG. The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.1196.203&1-5-13-7ALL. The port AID 1-5-13-7 must denote a VCG.

7. **`connAlias`** – Connection Alias
   This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83.
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes.
8. **`connStatus`** – Connection Status
   ConnStatus = pst&sst(&tst) Connection status is the overall status of the connection connID which may be different than the status of an individual *pathID* in that connection.

   connStatus is composed out of

   - pst (PrimaryState)
   - sst (SecondaryState)
   - tst (TertiaryState, optional).

   Up to two of the allowed values are simultaneously permitted as a sst/tst parameter. The two values must be separated by an ampersand (“&”). Individual status values are 7 characters or less
   Type: up to three &-separated value sets.

   Range *pst* values:

   - CON: Connected
   - DLT: Deleted
   - NOCON: Not Connected

- MAINT: Maintenence Mode
- UNKNOWN: Conn state unknown.

Range *sst* & *tst* values

- OK: Conn good
- FAILED: Conn failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- PREMPT: Path pre-empted
- SETUPFL: Connection SetUp Failed
- FL-NVM: Not saved in NVM
- WTORV: Waiting to revert path
- EPRTFL: EDGE ort Failed
- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection)
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection)

Range *tst* values

- DEGRADED: Non-revertive restoration was not possible if the worker or protection path of a permanent 1+1 protected connection fails

9. **`pdir`** – Path Direction Topology
   Uni-direction A-Z (1WAY) or Bi-direction – (2WAY).
   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default), 2WAY

10. **`protectionClass`** – Path Protection Class
    This parameter defines the type of restoration protection assigned to the path connection during path setup or after path modification.
    Range:
    - UNPROT (path from source to destination is un-protected)
    - AUTONR (path from source to destination is a non-revertive auto-reroute path connection)
    - N1PLUS1 (paths from source to destination is set up as a network 1+1 connection)
    - AUTORV (path from source to destination is a revertive auto-reroute path connection)
    - YCONN (paths from the same source to two different destinations are set up as a bridged connection)
    - MBYNWRK (working connection in an M:N connection group)
    - MBYNPRT (protecting connection in an M:N connection group)
    - P1PLUS1NR (paths from source to destination are set up as a 1+1 protected non-revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)
    - P1PLUS1NRRV (paths from source to destination are set up as a 1+1 protected revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)
11. **`bandwidth`** – Requested bandwidth/signalRate
    ONNS bandwidth is equal to or greater than the minimum cross connect rate (*minXCrate*) of the port type. Note: Only SONET switching levels in enumeration. For actual cross-connections ONNS adapts to SDH formats automatically if the ingress port is SDH.
    Type: Value set.
    Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
    These values are defined for setting up adaptive ONNS connections of the corresponding bandwidth. Specific restrictions wrt. the setup of adaptive ONNS connections can be found in “Error responses” (p. 3-1739).
12. **`maxDelay`** – Maximum transport delay
    Maximum transport delay through all nodes in the path. Total path delay must not exceed this value.
    Range: 0 (default), is used to select no delay limit; >0, maximum transport delay in milliseconds.
    This are an optional output parameter in the **`ENT-PATH`** command reply.

13. **`srvID`** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.
14. **`nodeListUsage1`** – Node List Usage 1
    For connection setup NN used/uses the list of nodes included in the *nodeList1* array block in a manner specified by the *nodeListUsage1* variable. Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command.
    An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent *nodeID* will be ignored in EXCLUDE *nodeList1* during path setup. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
    Range:
    - EXPLICIT, *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.)
    - EXCLUDE, nodes in the *nodeList1* must all be excluded from the path created. In the case when the *protectionClass* is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the *nodeList1* below also apply to the second or restoration path.

    This are an optional output parameter in the **`ENT-PATH`** command reply.
15. **`nodeList1`** – Node List 1
    When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidx]).
    When *nodeListUsage1* = EXCLUDE: List of nodes that must all be excluded from both paths. The *nodeID* values are shown in the from: nid1([&nid2& …&nidx]).
    If the preceding required *nodeListUsage1* field is not supplied, this field must not be present. If user specifies empty EXCLUDE *nodeList1*, the connection should use indirect path from source node to destination node (i.e., at least one intermediate node is needed). If user specifies empty EXPLICIT *nodeList1*, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
    Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
    Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
    Range: nid1([&nidx])* or empty list (where X <= 8 when *nodeListUsage1* = EXPLICIT) (where X <= 15 when *nodeListUsage1* = EXCLUSIVE)
    This are an optional output parameter in the **`ENT-PATH`** command reply.

16. **`nodeListUsage2`**
For connection setup NN used/uses the list of nodes included in the *nodeList2* array block in a manner specified by the *nodeListUsage2* variable. Both, *nodeListUsage2* and *nodeList2*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
This parameter shall also be allowed when *protectionClass* is P1PLUS1NR.
Type: Value set.
Range: EXPLICIT, *nodeList2* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.).
This are an optional output parameter in the **`ENT-PATH`** command reply.This are an optional output parameter in the **`ENT-PATH`** command reply.This are an optional output parameter in the **`ENT-PATH`** command reply.
This are an optional output parameter in the **`ENT-PATH`** command reply.
17. **`nodeList2`**
When nodeListUsage2=EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The nodeID values are shown in the form: nid1([[&nid2& …&nidx]]).
If the preceding required *nodeListUsage2* field is not supplied, this field must not be present. If user specifies empty EXPLICIT *nodeList2*, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
This parameter shall also be allowed when *protectionClass* is P1PLUS1NR.
Type: &-separated list of *nodeIDs*. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
Range: nid1([&nidX])* or empty list where X <= 8 when *nodeListUsage1* = EXPLICIT.
This are an optional output parameter in the **`ENT-PATH`** command reply.
18. **`srlgExcludeList`**
List of Shared Risk Link Group identifiers that are to be used to exclude links from the connection path(s) being created.
Type: &-separated list of numeric strings | empty | 0
Example: 33&49&64
Range: up to 10 numeric strings, each in the range 1 – 65535. The value 0 (default) or an empty field indicates no constraint are applied.
This are an optional output parameter in the **`ENT-PATH`** command reply.

19. **`linkIncludeID`** – VPN Identifier
    A connection constraint link property such as *linkIncludeID* that must be valid to include the link in the connection path(s) to be created. *linkIncludeID* is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
    Range: 1 – 65535. 0 (default) or an empty field indicates no VPN constraint is applied.
    Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.
    This are an optional output parameter in the **`ENT-PATH`** command reply.
20. **`srcNID`** – Source nodeID
    ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
    Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
21. **`srcPortID`** – Source portID
    ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
    Value: dash-separated list of 5 numeric strings
    Example: 1-1-3-1-4
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
    Range for protecting connections in M:N connection groups: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#)
22. **`destNID`** – Destination nodeID
    ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
    Type: Four dot separated numeric strings. 0 – 255 for each numeric string.
23. **`destPortID`** – Destination PortID
    ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
    Value: dash-separated list of 5 numeric strings
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
    Range for protecting connections in M:N connection groups: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#)

24. **`pathID1`** – Path ID 1
 The PathID of the 1st path that was created.
 Type: character string of length up to 20.
 Range: nodeID-Index: nodeID – four dot separated numeric strings, each string 0 – 255; Index – Numeric string 0-65535
 The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
 Example: 29.13.196.203-65111
25. **`pathState1`** – Path State 1
 Path State1 = *pst*& *sst*[*&tst*]. pathState1 represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.
 PathState1 is composed out of *pst* (PrimaryState), *sst* (SecondaryState) and *tst* (TertiaryState, optional) Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
 Type: up to three &-separated value sets.
 Range *pst* values:
 - CON: Connected
 - NOCON: Not Connected
 - DLT: Deleted
 - MAINT: Maintenance mode.

 Range *sst* & *tst* values
 - OK: Conn good
 - FAILED-PATH: failed
 - SETUP: Conn SetUp occurring
 - TRDOWN: Conn teardown occurring
 - NO-IN: Conn failed, signal in bad
 - RSTOR: path restore occurring
 - RSTRD: Conn restored
 - RSTRFL: Conn restore failed
 - DLT-FL: Delete operation Failed
 - UNKNOWN: Conn state unknown
 - WTORSTR: Waiting to restore path
 - SETUPFL: Connection SetUp Failed
 - RVRTFL: Conn revert failed
 - PREMPTD: Preempted
 - WTORV: Waiting to revert path
 - BRDGD: Path bridged
 - EPRTFL: EDGE Port Failed

- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

26. **`pNodeList1`** – Path Node List 1
The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
27. **`pathID2`** – Path ID 2
The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index: nodeID – four dot separated numeric strings, each string 0 – 255; Index – Numeric string 0-65535.
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111
This are an optional output parameter in the **`ENT-PATH`** command reply.
28. **`pathState2`** – Path State 2
pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values

- OK: Conn good
- FAILED-PATH: failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring

- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.
This are an optional output parameter in the **`ENT-PATH`** command reply.

29. **`pNodeList2`** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
This are an optional output parameter in the **`ENT-PATH`** command reply.

30. **`brDestNID`** – Bridged Destination NodeID
ONNS bridged destination nodeID. Together with the following brDestPortID field defines the second associated Egress connPoint that connects the destination node to the edge/gateway switch or client router in case of protectionClass YCONN.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

31. **`brDestPortiD`** – Bridged Destination PortID
ONNS bridged destination node egress *portID* in case *protectionClass* = YCONN. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the brDestNID egress connPoint. 4.1; 5; 6
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).

32. **`tribAID`** – Tributary AID
The *connPoint* on the reporting port of the cross-connection *connPoint* pair. The *connPtA* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # .
Type: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Special values for slot/pack, port, and tributary
Slot/Pack # #
Port # #
Tributary # #
Value "#" for slot/pack, port, and tributary indicates default tributary AID used for reporting the open end in not existing cross-connections in the A-node or Z-node for a protecting NN connection of an M:N protected connection group.
Slot/Pack # *
Port # *
Tributary # *
Value "*" for slot/pack, port, and tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter *connPtA*) is allocated by one or more LO traditional cross-connections.
Slot/Pack # @
Port # @
Tributary # @
Value "@" for slot/pack, port, and tributary indicates that the corresponding HO input and/or output tributary specified by *connPtA* is involved as EDGE-prt tributary in an OUC.
Slot/Pack # $
Port # $
Tributary # $
Value "$" for slot/pack, port, and tributary indicates a temporary reservation of an output tributary of an ENNI port needed for a network connection setup.

Spec block parameters

Spec block parameters are repeated in the output when explicitly supplied in the input. Multiple parameters are separated by &.

1. **`connMode`** – Connection Mode
   This parameter specifies whether revertive or non-revertive mode is used as operation mode of the auto-reroute or 1+1 protected connection.
   This parameter can be used with the ENT-PATH-AUTO and ENT-PATH-N1PLUS1 command. This parameter can be used with the ENT-PATH-P1PLUS1NR command to specify, whether revertive or non-revertive mode is used as operation mode of the 1+1 protection of the permanent 1+1 protected connection.
   Type: Alphanumeric value set.

   Range:

   - NONREVERTIVE (default)
   - REVERTIVE
2. **`rstrPriority`** – Restoration Priority
   This parameter describes the restoration priority of the connection.
   Range: 0 (default) – 7. Lower values represent higher priority.
3. **`disjointness`**
   The actual level of disjointness of the calculated and established connection. The parameter is not filled when protectionClass is UNPROT.
   Type: Alphanumeric value set.

   Range:

   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.
4. **`grpSetupType` –** grouped Setup Type
   The grouped Setup Type specifies, whether for all NN connections of a grouped setup the total cost of all connections or the number of routes used by the connections are minimized.

   Both alternatives will lead to equal results in the following cases:

   - Only one connection is setup with a grouped setup
   - There is enough bandwidth available via the cheapest route for all connections of a grouped setup.

   Type: Alphanumeric value set.
   Range: MINCOST (default): the total cost for all connections of the grouped setup is minimized. MINROUTES: the number of routes used by the connections of the grouped setup is minimized.

5. **`protRatio`** – Protection ratio
   For each working or protecting connection of an M:N protected connection group this parameter will report the concrete protection ratio as specified on creation of the group. For connections not being member of an M:N protected connection group, this parameter is not reported at all.
   Type: Alphanumeric value set
   Range. BY1 (ratio of 1 to 1 in number of protecting and working connections), 1BY2 (ratio of 1 to 2 in number of protecting and working connections), 1BY3 (ratio of 1 to 3 in number of protecting and working connections).
6. **`htmr`** – Hold-Off Timer
   This parameter applies to 1+1 protected and permanent 1+1 protected connections. This parameter specifies the value of the hold-off timer for 1+1 protection switching.
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms]
   Range:
   For high order path protection groups, the range is:
   - 0 .. 99 ms in 1 ms steps and
   - 100 .. 10000 ms in100 ms steps

   For low order path protection groups, the range is:
   - 0, 25, 50, 75 ms (thus in steps of 25 ms) and
   - 100 .. 10000 ms in 100 ms steps.

   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
   Default: 0
7. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait-to-restore time for revertive 1+1 protection switching.
   Parameter does not apply to a 1+1 protected connection revertive.
   Parameter applies to a permanent 1+1 protected connection with revertive SNCP/UPSR protection.
   Type: numeric string [minute].
   Range: 0 – 60.
   Default:
   None.
   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block. Default Value: 0

8. **switchBack** – revertive restoration path switch back
   This parameter describes how a revertive restoration path is switched back to the original path.
   Type: Alphanumeric value set.
   Range:
   - AUTO: path is automatically switched back after wait to revert time.
   - MAN: path has to be manually switched back triggered by the TL1 command **ED-PATH-ROLL**.

   Default: AUTO.

Example input/output

Input/output example for ENT-PATH-UNPROT

The following is an example shows a successful execution of the **ENT-PATH-UNPROT** command:

```
ENT-PATH-UNPROT:LT-NN-789:snn:653456::1351591:1WAY:sts48:105:135.13.196.203:1-5-13-4
      -1:135.13.196.123:1-2-5-5-
      1:exclude,135.13.42.3&135.213.242.103&135.13.198.22:33&49&64,5::connAlias=AmericanExp-NY-

   LT-NN-789 2002-07-18 08:00:00
M  653456 COMPLD
   ”ConnID=135.13.196.203&1-5-13-4-1&1,ConnStatus=CON&OK,1WAY,UNPROT,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,

      MaxDelay=105,SrvID=1351591,Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,SrlgExclude
   SrcNID=135.13.196.203,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.13.196.203-43216,State=CON&OK,
   NodeList=135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123”
;
```

Input/output example for ENT-PATH-AUTO

The following example shows a successful completion of an **ENT-PATH-AUTO** command for a bidirectional path setup with three nodes in the node exclusion list. The response contains the list of nodes traversed by the new path.

```
ENT-PATH-AUTO:LT-NN-789:snn:973456::1352463:2WAY:sts48:65:135.13.196.203:1-5-13-4-
      1:135.13.196.123:1-2-5-5-
      1:exclude,135.13.42.3&135.213.242.103&135.13.198.22:17&35&89,0::connMode=REVERTIVE,rstrPr

   LT-NN-789 2002-09-18 08:00:00
M  973456 COMPLD
   ”ConnID=135.13.196.203&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,AUTORV,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,
   MaxDelay=65,SrvID=1352463,Exclude=135.13.42.3&135.13.52.103&135.13.18.22,,
SrlgExclude=17&35&89,IncludeID=0,ConnMode=REVERTIVE&RstrPriority=1,,
   SrcNID=135.13.196.203,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.13.196.203-43216,State=CON&OK,
```

```
   NodeList=135.13.196.203&135.13.42.3,135.213.242.133&135.13.196.223:
   PathID=135.13.196.203-43231,State=NOCON,
   NodeList=135.13.196.203&135.13.22.3&135.13.24.3&135.13.44.93&135.13.196.123"
;
```

Input/output example for ENT-PATH-N1PLUS1

The example shows a successful completion of an **ENT-PATH-N1PLUS1** command for a bidirectional path setup with three nodes in the exclusion list. The response contains the list of nodes traversed by the new paths:

```
ENT-PATH-N1PLUS1:LT-NN-789:snn:93456::1351315:2WAY:sts48:65:135.13.196.203:1-5-13-4-
      1:135.13.196.123:1-2-5-5-
      1:Exclude,135.13.42.3&135.213.242.103&135.13.198.22:17&35&89&6,10::connAlias=AmericanE

   LT-NN-789 2002-09-18 08:00:00
M  93456 COMPLD
   "ConnID=135.13.196.203&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,N1PLUS1,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,

      MaxDelay=65,SrvID=1351315,Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,SrlgExclu
   SrcNID=135.13.196.203,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1:
   PathID=135.13.196.203-43216,State=CON&OK&ACT,
   NodeList=135.13.196.203&135.13.42.3&135.213.242.133&135.13.196.123:
   PathID=135.13.196.203-43231,State=CON&OK,
   NodeList=135.13.196.203&135.13.22.3&135.13.24.3&135.13.44.93&135.13.196.123"
;
```

Input/output example for ENT-PATH-YCONN

The example shows a successful completion of an **ENT-PATH-YCONN** command for a bidirectional path setup with three nodes in the exclusion list. The response contains the list of nodes traversed by the new paths:

```
ENT-PATH-YCONN:LT-NN-789:snn:93456::1351315:2WAY:sts48:65:135.13.196.203:1-5-13-4-
      1:135.13.196.123,135.13.196.234:1-2-5-5-1,1-3-4-4-
      2:Exclude,135.13.42.3&135.213.242.103&135.13.198.22:17&35&89&6,10::
      connAlias=AmericanExp-NY-BOS-ckt;

   LT-NN-789 2002-09-18 08:00:00
M  93456 COMPLD
   "ConnID=135.13.196.203&1-5-13-4-1&1,ConnStatus=CON&OK,2WAY,YCONN,STS48,
   ConnAlias=AmericanExp-NY-BOS-ckt,

      MaxDelay=65,SrvID=1351315,Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,SrlgExclu
   SrcNID=135.13.196.203,1-5-13-4-1,DestNID=135.13.196.123,1-2-5-5-1,brDestNID=135.
      13.196.234,1-3-4-4-2:
   PathID=135.13.196.203-43216,State=CON&OK &ACT ,
   NodeList=135.13.196.203&135.13.42.3&135.213.242.133&135.13.196.123:
   PathID=135.13.196.203-43231,State=CON&OK,
```

```
   NodeList=135.13.196.203&135.13.22.3&135.13.24.3&135.13.44.93&135.13.196.234"
;
```

Input/output example for ENT-PATH-MBYN

The example shows a successful completion of an **ENT-PATH-MBYN** command for a bidirectional path setup for 2 working connections to be 1:2 protected. The response contains all 3 connections setup (2 working and 1 protecting connections):

```
ENT-PATH-MBYN:LT-NN-789:snn:973456::1352463::2WAY:sts1:65:135.13.196.203:
1-1-13-4-1&2:135.13.196.123:1-1-5-5-5&6::,::
connAlias=AmericanExp-NY-BOS-ckt,protRatio=1BY2;

   LT-NN-789 2002-09-18 08:00:00
M  973456 COMPLD

   "ConnID=135.13.196.203&1-1-13-4-1&1,ConnStatus=CON&OK,2WAY,MBYNWRK,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_wrk_001,
   MaxDelay=65,SrvID=135246,,,,IncludeID=0,,,ProtRatio=1BY2,
   SrcNID=135.13.196.203,1-1-13-4-1,DestNID=135.13.196.123,1-1-5-5-5:
   PathID=135.13.196.203-43216,State=CON&OK,
   NodeList=135.13.196.203&135.13.42.3&135.213.242.133&135.13.196.223"

  "ConnID=135.13.196.203&1-1-13-4-3&2,ConnStatus=CON&OK,2WAY,MBYNWRK,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_wrk_002,
   MaxDelay=65,SrvID=135246,,,,IncludeID=0,,,ProtRatio=1BY2,
   SrcNID=135.13.196.203,1-1-13-4-3,DestNID=135.13.196.123,1-1-5-5-6:
   PathID=135.13.196.203-43217,State=CON&OK,
   NodeList=135.13.196.203&142.330.17.1&222.177.23.4&135.13.196.223"

   "ConnID=135.13.196.203&1-1-#-#-#&3,ConnStatus=CON&STBY,2WAY,MBYNPRT,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_prt_001,
   MaxDelay=65,SrvID=135246,,,,IncludeID=0,,,ProtRatio=1BY2,
   SrcNID=135.13.196.203,1-1-#-#-#,DestNID=135.13.196.123,1-1-#-#-#:
   PathID=135.13.196.203-43218,State=RSVD&OK,
   NodeList=135.13.196.203&222.111.22.2&141.235.1.17&135.13.196.223"
;
```

Input/output example for ENT-PATH-MBYN worker connection extension

The example shows a successful completion of an **ENT-PATH-MBYN** worker connection extension command for a bidirectional path setup to an existing M:N protection group.

```
ENT-PATH-MBYN:LT-NN-789:snn:973456::1352463:135.13.196.203&123::::
135.13.196.203:1-1-13-4-3:135.13.196.123:1-1-5-5-6::,::
connAlias=AmericanExp-NY-BOS-ckt;

   LT-NN-789 2002-09-18 08:00:00
M  973456 COMPLD
```

```
   "ConnID=135.13.196.203&1-1-13-4-1&3,ConnStatus=CON&OK,2WAY,MBYNWRK,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_wrk_003,
   MaxDelay=65,SrvID=135246,,,,IncludeID=0,,,ProtRatio=1BY2,
   SrcNID=135.13.196.203,1-1-13-4-3,DestNID=135.13.196.123,1-1-5-5-6:
   PathID=135.13.196.203-43216,State=CON&OK,
   NodeList=135.13.196.203&135.13.42.3&135.213.242.133&135.13.196.223"

   "ConnID=135.13.196.203&1-1-#-#-#&5,ConnStatus=CON&STBY,2WAY,MBYNPRT,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_prt_002,
   MaxDelay=65,SrvID=135246,,,,IncludeID=0,,,ProtRatio=1BY2,
   SrcNID=135.13.196.203,1-1-#-#-#,DestNID=135.13.196.123,1-1-#-#-#:
   PathID=135.13.196.203-43218,State=RSVD&OK,
   NodeList=135.13.196.203&222.111.22.2&141.235.1.17&135.13.196.223"
;
```

Error responses

Validation rule: Insufficient network resources

Error code:

`E001 – srvID – InSufficient Network Resources for connection setup`

Validation rule: Insufficient Network Resources for Alternate Route

If network resources for an alternate route of a restoration protected connection to be setup are insufficient, then the following error response is returned:

Error code:

`E004 – srvID – InSufficient Network Resources for Alternate Path`

The following list identifies conditions that will cause similar error responses:

Error code: `E012- SXCF – Status, XConn Failure caused path setup failure`

Error code:

`E013 – SSCF – Status, Signaling Communication link Failure (or abort) caused path operation failure`

Error code: `E014 – SXCT – Status, XConn Timeout`

Error code: `E016 – SHTO – Status, Host interface TimeOut`

Error code:

`E017 – SSTO – Status, Signaling Timeout caused path setup failure`

Error code: `E018 – SEPU – Status, EDGE Port Used up`

Error code:

`E022 – SPSF – Status, Path Setup Failure, caused by signal transmission failure at egress node`

E016 and E022 are only sent in notifications, not in command replies.

Error code: `E021 – SUNA – Status, port class UNI Not Allowed`

Validation rule: Maximum Delay Exceeded by calculated path

When the Maximum Transport Delay Exceeded, then the following error response is returned:

Error code: `E020 – srvID – Maximum Transport Delay Exceeded,errorData`

Validation rule: ENT-PATH syntax, parameter data invalid

When an invalid input parameter is used, then the following error response is returned:

Error code:
`E030 – srvID – Syntax, Parameter Data Invalid,invalid_parameter= fieldName`

The following list identifies conditions that will cause similar error responses:

Error code: `E032 – Input, Source node IDis invalid`

Error code: `E033 – Input, Node List has Invalid / unknown ID`

Error code: `E035 – This connection Alias Already Exists`

Validation rule: ENT-PATH syntax, invalid degree of disjointness value

When an invalid degree of disjointness value for protClass N1PLUS1 or YCONN is used, then the following error response is returned:

Error code: `E520 – IIDD – Input, Invalid Degree of Disjointness`

Validation rule: ENT-PATH syntax, data not consistent

When inconsistent input parameters are used, then the following error response is returned:

Error code: `E040 – srvID – Syntax, Data Not Consistent`

Validation rule: NN PortID parameter value is not valid

If the NE receives a connection request where the *portID* parameter value is not valid, then one of the following error response is returned:

Error code: `E081 – IIEP – Input, Invalid ingress / Egress Port,errorData`

The following list identifies conditions that will cause similar error responses:

Error code:
`E080 – IICP – Input, Invalid ConnPt, ID=<ASTN node ID>&<connPtID>`

Error code:
`E081 – IIEP – Input, Invalid ingress / Egress Port, ID=<ASTN node ID> &<connPtID>`

Error code:
`E082 – SCPU – Status, NNI connPoint / port Unavailable, ID=<ASTN node ID>&<connPtID>`

Error code:
`E083 – SCIU – Status, NNI connPoint in Use, XConn or reservation exists, ID=<ASTN node ID>&<connPtID>`

Error code:
`E084 – SHXC – Status, Host XConn on ConnPoint, ID=<ASTN node ID> &<connPtID>`

Error code E084 will also be generated in case of a normal (non-forced) loopback exists on the ConnPt, that is, a normal loopback is considered as a host cross-connection.

The following list identifies conditions that will cause error responses specific to ENT-PATH commands affecting VCG EDGE ports:

- Error code:
  `E052 – IIET – Input, Invalid EDGE port type combination with VCG connection setup`

The following list identifies conditions that will cause error responses specific to ENT-PATH commands to setup groups of connections:

- Error code:
  `E053 – IIGS – Input, Port type in srcPortID does not support group setup/SrcPortID`
- Error code:
  `E055 – IIPG – Input, Invalid protection type YCONN for connection groups`
- Error code:
  `E056 – IISD – Input, Grouped setup with inconsistent values of destPortID and srcPortID`

The following list identifies further conditions that will cause error responses specific to ENT-PATH commands wrt. the parameter grpSetupType in a grouped connection setup:

- Error code:
  `E057 – IIGT – Input, Parameter grpSetupType only allowed if protType is UNPROT`

The following list identifies conditions that will cause error responses specific to ENT-PATH commands to setup M:N protected groups of connections:

- Error code:
  `E058 – IISP – Input, Invalid Source Port type for M:N group setup`
- Error code:
  `E059 – IIDP – Input, Invalid Destination Port type for M:N group setup`
- Error code:
  `E060 – IPNE – Input, No EXPLICIT routing allowed for M:N group setup`
- Error code:
  `E061 – STSF – Status, Connection torn down due to M:N group setup failure`

The following list identifies further conditions that will cause error responses specific to ENT-PATH commands to setup M:N protected groups of connections:

- Error code: `E063 – INCG – Input, M:N groupID unknown`
  This error is responded if the specified M:N groupID for a connection extension unknown.

The following list identifies conditions that will cause error responses specific to ENT-PATH commands to add a working connection to a M:N protected group of connections:

- Error code:
  `E064 – INTL – Input, Number of source/destination Tributaries too Large`
  This error is responded if more than one source and/or destination tributary are provisioned to add a working connection to an existing M:N group.
- Error code:
  `E065 – IAWE – Input, number of Added Working connection Exceeded`
  This error is responded if the maximal number of working connections after adding working connections is exceeded.
- Error code:
  `E066 – IPNC – Input, Parameter Not allowed for M:N Connection extension`
  This error is responded if parameters like exclude SRLG, exclude nodes for a added working connection are provisioned.
- Error code: `E279 – SNRE – Status, NVM Record Exhausted`
  This failure condition is only valid until a variable NVM allocation scheme is implemented, that does not have any restrictions wrt the length of connection records to be stored.

Validation rule: Cross-connect connect failure

If the network element receives an **ENT-PATH** command, but when the request is failed due to cross-connect connect failure at a node during path setup, the following error response is returned:

```
   srcTid 2003-01-01 08:00:00
M  ctag DENY
   CFPE
   ″E160:srvID:\″XConn Connect Failure, Port does not exist or is TRADITIONAL\″,″
;
```

Error code:

`E160 – srvID – XConn Connect Failure, Port does not exist or is TRADITIONAL`

The following list identifies conditions that will cause similar error responses:

Error code:

`E161 – CFUT – XConn Connect Failure, Unexpected toplogy or related cross connect, ID=<ASTN node ID>&<connPtID1>&<connPtID2>`

Error code:

`E162 – CFXE – XConn Connect Failure, Cross connect already exists, ID= <ASTN node ID>&<connPtID1>&<connPtID2>`

Error code:

`E168 -CFPT – XConn Connect Failure, Invalid Access to Protection Tributary on 2-f BLSR/MS-SPRing EDGE port,ID=<ASTN node ID>&<connPtID>`

The following list identifies error conditions wrt. setting up of adaptive ONNS connections:

- Error code:
  `E170 – IIPA – Input, Invalid Protection type for Adaptive ONNS connection`
- Error code:
  `E171 – STMI – Status, Tributary Mode of ingress / egress Port Inconsistent with requested bandwidth, ID=<ASTN node ID>&<connPtID>`

Error response E171 will be raised in the following cases:

- It is tried to setup an adaptive mode ONNS connection (specified by a corresponding adaptive bandwidth value) on an EDGE port with fixed tributary mode
- It is tried to setup a fixed mode ONNS connection (specified by a corresponding fixed bandwidth value) on an EDGE port with adaptive tributary mode

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "ED-NNCNFG: Modify ONNS system configuration" (p. 3-450)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)

ENT-PROTN-GRP: Enter protection group

Protection groups are used in protection switching. A protection group is made up of one or more working (service) entities and a protection entity. In addition to the entities that comprise the protection group, the protection group has parameters that affect the characteristics of the protection group, such as whether or not the protection switching is revertive.

The **`ENT-PROTN-GRP`** command allows one to create a user-provisioned association of optical interface ports which is used in protection. Protection groups also determine the set of tributaries from and to which the user can provision cross-connections. If a circuit pack contains more than one optical port, the user can provision the type of protection for each port independently of the other ports. Unprotected operation is supported for optical interface ports that are not in a protection group.

When creating a protection group, the following must be true (only a subset of the validation rules is described here – see corresponding error response section):

- the individual ports must not already be in another protection group.
- there are no existing cross-connections on the ports.
- each port in a protection group must be on a different circuit pack.

Some types of protection groups are created automatically by the system or with another stimuli. ENT-PROTN-GRP cannot be used to create the following types of protection groups:

- switching equipment protection (created as part of slot assignment or pack auto-creation)
- controller equipment protection (created as part of slot assignment or pack auto-creation)
- path protection group (created as part of ENT-CRS for path-protected cross-connection).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

Command syntax: **`ENT-PROTN-GRP`**:*tid*::*ctag*::*protype*:*spec_block*;

Input parameters

Command parameters (equipment protection group)

The command **`ENT-PROTN-GRP`** is applicable for EP155, EP51, LOXC, TransMUX, and ES64U equipment protection.

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`ctag`** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`protype`** – Protection Type
   The parameter specifies the type of the protection group.
   Type: Alphanumeric value set.
   Range:
   - EQPTEP155 (if aid is EP155ProtGroupAID)
   - EQPTEP51 (if aid is EP51ProtGroupAID)
   - EQPTLOXC (if aid is LOXCProtGroupAID)
   - EQPTTMUX (if aid is TMUXProtGroupAID)
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

   Default: None.

Command parameters (multiplex section/line protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`ctag`** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`protype`** – Protection Type
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later)
   Type: Value set.
   Range: 1+1 or 1xNOPT

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`ctag`** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F

Spec block parameters (equipment protection group)

1. **`pgid`** – protection group Id
   This parameter specifies the protection group Id as component of the protection group AID (AID field *TransProtGrpID*). For values less than 100, the leading zeros are optional. For example, either 007 or 7 is acceptable. If the user omits the leading zeros, the system will add them.
   Type: Numeric string.
   Range: 01 .. 99 (if *protype* is EQPTEP155, EQPTEP51, EQPTTMUX, or EQPTES64U); 01(if *protype* is EQPTLOXC).
2. **`protn`** – Protection Unit
   The parameter specifies the protection circuit pack AID being a member of the 1+1 equipment protection group.
   Type: AID
   Default: For switching protection group instance: CpAID related to slot 10. For controller protection group instance: CpAID related to slot 31. For EP155 protection group instance: – If no cross-connections are assigned to one of the two EP155 cards, the user can choose it by the **`ENT-PROT-GRP`** command Default protection unit is the right card. – If cross-connections are assigned to one of the cards, the other card must become the protection unit. This means that the creation (ENT-PROTN-GRP) is rejected, if the user selects the card with cross-connections as protection card.
3. **`wkg`** – Worker Unit
   This parameter specifies the worker circuit pack AID being a member of the 1+1 equipment protection group.
   Type: AID
   Range: CpAID
   Default: For switching protection group instance: CpAID related to slot 9. For controller protection group instance: CpAID related to slot 11. For EP155 protection group instance: – If no cross-connections are assigned to one of the two

EP155 cards, the user can choose it by the **`ENT-PROT-GRP`** command Default worker unit is the left card. – If cross-connections are assigned to one of the cards, this card must become the worker unit. This means that the creation (ENT-PROTN-GRP) is rejected, if the user selects the card with no cross-connections as worker card.

Spec block parameters (multiplex section/line protection group)

1. **`pgid`** – protection group Id
   This parameter specifies the protection group Id as component of the protection group AID (AID field *TransProtGrpID*). For values less than 100, the leading zeros are optional. For example, either 007 or 7 is acceptable. If the user omits the leading zeros, the system will add them.
   Type: Numeric string.
   Range: 1 … 999
2. **`protn`** – Protection Unit
   This parameter specifies the protection port AID being a member of the line protection/ MS protection group.
   Type: AID
   Range: SPortAID
3. **`protocol`** – APS Protocol
   This parameter specifies the protocol used by the protection state machine.
   The value 1+1_UNI is used for 1+1 unidirectional switching.
   The value 1x***N***_BIDIR is used for 1:1 bidirectional switching (***N*** = 1) (only R3.0 and later).
   The value 1+1_OPTM is used for 1+1 optimized, bidirectional switching (only R3.0 and later). The value 1+1_BIDIR is used for 1+1 bidirectional switching (only R4.0 and later).
   The value 1+1_BIDIR is used for 1+1 bidirectional switching (only R4.0 and later).
   The value 1+1_UNI shall be used for 1+1 unidirectional switching.
   The value 1x***N***_BIDIR shall be used for 1:N bidirectional switching (***N*** = 1..14).
   The value 1+1_OPTM shall be used for 1+1 optimized, bidirectional switching.
   The value 1+1_BIDIR shall be used for 1+1 bidirectional switching.
   Note that “OPTIMIZED” describes rather the protection type than a protocol. Only for compatibility reasons of TL1, the parameter *protocol* is used for its selection.
   Type: Alphanumeric value set.
   Range: 1+1_UNI, 1xN_BIDIR, 1+1_OPT.
   In mind the ***N*** in 1x***N***_BIDIR is considered to be always equal “1”. The value 1+1_OPTM implicitly specifies bidirectional operation mode.
   Default: 1+1_UNI, in case of protype = 1+1; 1xN_BIDIR, in case of protype = 1xNOPT (that means protection type 1:1).
4. **`rme`** – Protection Reversion
   The parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group. The value ENABLE is used if the protection group operates in revertive mode. The value DISABLE is used if the protection group operates in non-revertive mode.

Type: Alphanumeric value set.
Range: DISABLE in case of protype = 1+1; ENABLE in case of protype = 1xNOPT.
Default: DISABLE, in case of protype = 1+1; ENABLE, in case of protype = 1xNOPT (i.e. protection type 1:1)

5. **`wkg`** – Worker Unit
   This parameter specifies the worker port AID being a member of the 1+1 line/MS protection group.
   Type: AID
   Range: SPortAID

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value manually with ED-MAP-RING and supply the ring map information to each node on the ring for the BLSR protocol to operate. It is necessary to activate this user-provisioned information with SET-MAP-RING. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
   Type: Alphanumeric string.
   Range: ENABLE, DISABLE.
   Default: ENABLE.
2. **`asmc`** – Automatic Squelch Map Calculation
   If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
   Type: Alphanumeric string
   Range: ENABLE, DISABLE (default)
3. **`east`** – East Port
   This parameter specifies the port being a member in the role of the east port for a 2-fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: SPortAID
4. **`pgid`** – protection group Id
   The parameter specifies the protection group Id as component of the protection group AID (AID field TransProtGrpID). For values less than 100, the leading zeros are optional. For example, either 007 or 7 is acceptable. If the user omits the leading zeros, the system will add them.
   Type: Numeric string.
   Range: 1 – 999

5. **`rcamode`** – Ring Circuit Alarm Mode
This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE
6. **`rid`** – Ring Identification
Option: The parameter is a mandatory parameter.
This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE.
7. **`west`** – West Port
This parameter specifies the port being a member in the role of the west port for a 2-fibre MS-SPRing/BLSR protection group.
Type: AID.
Range: SPortAID.
Default: None.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value and supply the ring map information to each node on the ring for the BLSR protocol to operate. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.
2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: Alphanumeric string
Range: ENABLE, DISABLE (default)
3. **`eapsprotn`** – Incoming East APS Protection Path
This parameter is derived from the incoming K2 overhead byte of the east port.

It indicates the protection path as follows:

- 0: short path code (S)
- 1: long path code (L)

The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric value set.
Range: 0, 1, UNAVAILABLE.
Default: 0.

4. **`eastwkg`** – East Worker Port
   This parameter specifies the port being a member in the role of the east worker port for a 4-fibre MS-SPRing/BLSR protection group.
   Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID.
   Range: In case of an asymmetric ring: NOTUSED. In all other cases: SPortAID.
5. **`pgid`** – protection group Id
   The parameter specifies the protection group Id as component of the protection group AID (AID field TransProtGrpID). For values less than 100, the leading zeros are optional. For example, either 007 or 7 is acceptable. If the user omits the leading zeros, the system will add them.
   Type: Numeric string.
   Range: 1 – 999
6. **`protnmode`** – Protection Switch Mode
   The parameter specifies the protection switch mode. The value LOOPBACK is used for the classical ring application. The value SHORTENED is used for transmission delay sensitive applications. The value TOPEX (only R3.0 – R3.2 and R8.0.1 and later) is used for the so called transoceanic application, which allows extra traffic, improves restorability and improves delay as well as in SHORTENED mode.
   In case of a ring switch … LOOPBACK performs the bridge and switch function in that nodes which are most close to the failed span. SHORTENED performs the bridge function at the add node repectively the switch function at the drop node of each ring circuit. It is an application only similar to the transoceanic protocol. TOPEX allows, compared to SHORTENED mode, additionally the traffic restoration of not affected spans. It is the transoceanic protocol application.
   Consistent Protection Switch Mode Setting: The Protection Switch Mode must be set consistently on all nodes in the ring. E.g. for all UNITE NEs in a TOP+EX ring the parameter protnmode must be set to TOPEX. In a ring with inconsistent Protection Switch Mode, there will be no protection switching. The UNITE NEs will raise an Inconsistent Ring Protection Mode alarm.
   Type: Alphanumeric value set.
   Range: LOOPBACK, SHORTENED, TOPEX.
   Default: LOOPBACK.

7. **rcamode** – Ring Circuit Alarm Mode
   This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
8. **rid** – Ring Identification
   Option: The parameter is a mandatory parameter.
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE
9. **westprotn** – West Protection Port
   This parameter specifies the port being a member in the role of the west protection port for a 4-fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: SPortAID
10. **westwkg** – West Worker Port
    This parameter specifies the port being a member in the role of the west worker port for a 4-fibre MS-SPRing/BLSR protection group
    Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID
    Range: In case of an asymmetric ring: NOTUSED; in all other cases: SPortAID.

Output format

After receiving the **ENT-PROTN-GRP** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response.

Detailed behavior description

EP155 equipment protection group

The command establishes/creates a protection group of type as specified by means of parameter protype. If protype equals ’EQPTEP155’, then one instance of EP155 protection group will be created. The creation of the protection group is reported by means of an ENT-PROTN-GRP notification. The initial protection state information is reported via protection switch notification.

Port impact due to EP155 protection group - port deletion

All ports related to the protection card of the protection group is deleted. This deletion is reported by means of: A set of DLT-STM1E notifications if the protection port is an SDH port. The deletion of the objects leads to a clearance of pending alarms for these objects.

Port impact due to EP155 protection group - PM impact due to port deletion

PM monitoring for the deleted ports are stopped and current and history bins are deleted.

Tributary impact due to EP155 protection group - tributary deletion

All tributaries related to the protection port of the protection group is deleted. This deletion is reported by means of: A set of DLT-VC3 notifications if the protection port is an SDH port.

The number of tributaries to be deleted is derived from the port rate. The deletion of the objects leads to a clearance of pending alarms for these objects.

For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to 1&&-<N> with N is maximal legal STS-1/AU-3 number

Tributary impact due to EP155 protection group - AID range for group notification

See STS-1/AU-3 Tributaries AID w.r.t. range for legal STS-1/AU-3 number. Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192

PM impact due to tributary deletion

PM monitoring for the deleted tributaries are stopped and current and history bins are deleted.

EP51 equipment protection group

The command shall establish/create a protection group of type as specified by means of parameter protype. If protype equals ’EQPTEP51’, then one instance of EP51 protection group will be created.

The creation of the protection group is reported by means of an ENT-PROTN-GRP notification. The initial protection state information is reported via protection switch notification.

Port impact due to EP51 protection group - port deletion

All ports related to the protection pack of the protection group is deleted.

This deletion is reported by means of:

- A set of DLT-T3 notifications for the DS3 ports on the protection pack
- A set of DLT-EC1 notifications for the EC1 ports on the protection pack

The deletion of the objects leads to a clearance of pending alarms for these objects.

Port impact due to EP51 protection group - PM impact due to port deletion

PM monitoring for the deleted ports are stopped and current and history bins are deleted.

Tributary impact due to EP51 protection group - tributary deletion

All tributaries related to the protection port of the protection group is deleted.

This deletion is reported by means of:

- A DLT-PTF-STS notification for each deleted DS3 port of the protection pack
- A DLT-STS1 notification for each deleted EC1 port of the protection pack

The deletion of the objects leads to a clearance of pending alarms for these objects.

Tributary impact due to EP51 protection group - PM impact due to tributary deletion

PM monitoring for the deleted tributaries are stopped and current and history bins are deleted.

LOXC equipment protection group

The command establishes/creates a protection group for LOXC packs if *protype* is set to EQPTLOXC.

The creation of this protection group is reported by means of an **`ENT-PROTN-GRP`** notification. The initial protection state information has to be reported via protection switch notification.

The existence of transmission entities is not influenced by the creation of the LOXC EQP group since they will be only controlled by lower-order cross-connection commands.

LOXC – update affected entities

When an LOXC equipment protection group is created, then the lower-order switch capacity of the protection unit is set to the same size as for the related worker unit. In addition, the LO tributary standard of the protection unit is set to the value of the related worker unit. This is notified by means of an **ED-EQPT** notification.

TransMUX equipment protection group

The command establishes/creates a protection group for TransMUX packs if *protype* is set to EQPTTMUX. The creation of this protection group is reported by means of an **ENT-PROTN-GRP** notification. The initial protection state information is reported via protection switch notification.

TransMUX – update affected entities

When a TransMUX equipment protection group is created, then the TransMUX capacity of the protection unit is set to the same size as for the related worker unit. This is notified by means of an **ED-EQPT** notification.

ES64U equipment protection group

The command establishes/creates a protection group for ES64U packs if *protype* is set to EQPTES64U. The creation of this protection group is reported by means of an **ENT-PROTN-GRP** notification. The initial protection state information is reported via protection switch notification.

ES64U – update affected entities

When an ES64U equipment protection group is created, then the ES64U capacity of the protection unit is set to the same size as for the related worker unit and the ES64U IP Address *dcn_ipad* is overtaken from worker to protection pack. This is notified by means of an **ED-EQPT** notification.

1+1 and 2F protection group

The command establishes/creates a protection group of type as specified by means of parameter *protype*. If *protype* equals ’1+1’, then one instance of 1+1 MS/line protection group will be created. If *protype* equals ’2F’, then one instance of type 2-fibre MS-SPRing/BLSR protection group will be created.

The creation of the protection group is reported by means of an **ENT-PROTN-GRP** notification. The initial protection state information is reported via protection switch notification.

1+1 protection group

All tributaries related to the protection port of the protection group is deleted if an MSP 1+1 is entered.

- A set of DLT-VC3 notifications if the protection port is an SDH port
- A set of DLT-STS1 notifications if the protection port is an SONET port

The number of tributaries to be deleted is derived from the port rate. The deletion of the objects leads to a clearance of pending alarms for these objects. For efficiency reasons the notifications for tributaries are sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to 1&&-<N> with N is maximal legal STS-1/AU-3 number.

Example: The AID field in a group notification for a tributaries related to port 1 for a OP10 pack in slot 1 looks like: 1-1-#-#-1-1-1&&-192.

4 F protection group

The command establishes/creates a protection group of type as specified by means of parameter *protype*. If protype equals 4F', then one instance of type 4 fibre MS-SPRing/BLSR protection group will be created.

If east side parameters (worker and protection) and west side (worker and protection) is specified in the command request, then a 2-sided MS-SPRing/BLSR protection group is created.

The creation of the protection group is reported by means of an **`ENT-PROTN-GRP`** notification. The initial protection state information is reported via protection switch notification.

1+1 Protection Group: Tributary impact due to MSP group - tributary deletion of MSP 1+1 protected connections

All tributaries related to the protection port of the protection group are deleted if an MSP 1+1 is entered:

- a set of DLT-VC3 notifications if the protection port is an SDH port
- a set of DLT-STS1 notifications if the protection port is an SONET port

The number of tributaries to be deleted are derived from the port rate.

The deletion of the objects leads to a clearance of pending alarms for these objects.

1+1 Protection Group: Tributary impact due to MSP group - group notification

The notifications for the affected tributaries are a group notification. See definition in detailed behavior description section of ENT-EQPT.

1+1 Protection Group: Tributary impact due to MSP group - PM impact due to tributary deletion

PM monitoring for the deleted tributaries is stopped and current and history bins are deleted.

Example input/output

Equipment protection group

The following example creates equipment protection group information:

```
ENT-PROTN-GRP:LT-UNITE::123456::EQPTEP155:wkg=1-1-#-#-21-CP protn=1-1-#-#-22-
      CP,pgid=01;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Multiplex section/line protection group 1+1

The following example creates 1+1 multiplex section/line protection group information:

```
ENT-PROTN-GRP:LT-UNITE::123456::1+1:wkg=1-1-#-#-13-1,protn=1-1-#-#-14-1,
pgid=001;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

2 fibre MS-SPRing/BLSR protection group

The following example shows the use of **ENT-PROTN-GRP** command to create a 2F protection group:

```
ENT-PROTN-GRP:LT-UNITE::123456::2F:east=1-1-#-#-10-1,west=1-1-#-#-08-
      1,pgid=003,rid=”denver21”;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

4 fibre MS-SPRing/BLSR protection group

The following example shows the use of **ENT-PROTN-GRP** command to create a 4F protection group:

```
ENT-PROTN-GRP:LT-UNITE::123456::4F:eastwkg=1-1-#-#-10-1, eastprotn=1-1-#-#-11-
      1,westwkg=1-1-#-#-08-1,westprotn=1-1-#-#-09-1,pgid=003,rid=”denver21”;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses (EP*n* equipment protection group)

Validation rule: EP*n* pack already used in other equipment protection group

The creation of an EP***n*** equipment protection group will be denied if the specified worker or protection pack is already a member of another EP***n*** equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: No HO/LO cross-connections, conversion cross-connections or substructure allowed

The creation of the EP***n*** Equipment Protection group is rejected, if HO or LO cross-connections, conversion cross-connections or lower order substructure exist on the EP***n*** pack, which was specified as protection pack by the command.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: Transparent DCC connections

Check of transparent DCC connections involving the circuit pack to be proposed as protection pack.

Error code:

`SNVS – Status, Not in Valid State, Transparent DCC connections are existing at protection pack`

Validation rule: Not used as timing reference

Check of that none of the ports of the proposed protection circuit pack is used as timing reference

Error code:

`SNVS – Status, Not in Valid State, at least one port of the protection pack is used as timing reference`

Validation rule: Not used for OW

Check of that no connection to a OW port of the proposed protection circuit pack exists.

Error code:

`SNVS – Status, Not in Valid State, connection to OW port of the protection pack exists`

Validation rule: No XC loopbacks allowed

Check of XC loopbacks involving the circuit pack to be proposed as protection pack.

Error code:

`SNVS – Status, Not in Valid State, XC loopbacks still exist at the protection pack`

Validation rule: No facility loopbacks allowed

Check of facility loopbacks involving the circuit pack to be proposed as protection pack.

Error code:

`SNVS – Status, Not in Valid State, facility loopbacks still exist at the protection pack`

Validation rule: No test access session

Tributaries related to the circuit pack to be proposed as protection pack must not be used in a test access session either as tributary under test or as test access tributary.

Error code:
`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: No port in *portClass* INNI

The creation of the EP155 Equipment Protection group is rejected, if at least one of the affected ports in in *portClass* INNI.

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: No port on protection pack in *portClass* EDGE

The creation of the EP155 Equipment Protection group is rejected, if at least one of the affected ports on the protection pack is in *portClass* EDGE.

Error code:
`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Validation rule: No port on protection pack in portClass UNI

The creation of the EP155 Equipment Protection group is rejected, if at least one of the affected ports on the protection pack is in *portClass* UNI.

Error code:
`SNVS – Status, Not in Valid State, Port is still in use by ONNS`

Error responses (LOXC Equipment Protection Group)

Validation rule: LOXC switch capacity and unit roles

The creation of the LOXC Equipment Protection group is rejected, if the worker unit has a low order switch capacity of 0 GBit/s or if the protection unit has a low order switch capacity greater than 0 GBit/s.

Error code:
`SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

This is not an exclusive “or”. If one condition is true, the error will be sent.

Other checks are not necessary. Cross-connections, loopbacks, test access and other configurations may have been applied to the worker unit while establishment of the LOXC equipment protection group.

Validation rule: Worker and protection LOXC must be of same type

The creation of the LOXC Equipment Protection group is rejected, if the worker LOXC type is different from the protection LOXC type.

Error code:

`SNVS – Status, Not in Valid State, worker and protection LOXC must be of same type`

Error responses (TransMUX protection groups)

Validation rule: TransMUX pack already used in other equipment protection group

The creation of an TransMUX equipment protection group will be denied if the specified worker or protection pack is already a member of another TransMUX equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: TransMUX switch capacity and unit roles

The creation of the TransMUX Equipment Protection group is rejected, if the worker unit has a TransMUX switch capacity of 0 GBit/s or if the protection unit has a TransMUX switch capacity greater than 0 GBit/s.

Error code:

`SNVS – Status, Not in Valid State, TMUX switch capacity invalid`

The creation of a TransMUX equipment protection group is only allowed, if the worker TransMUX pack has *tmuxcap* > 0 and the protection pack has *tmuxcap* = 0.

Validation rule: Slot restrictions for TransMUX equipment protection group

The creation of a TransMUX equipment protection group will only be accepted if the worker pack is plugged in a valid worker TransMUX slot and the protection unit is plugged in the adjacent slot to the right of the worker unit. In all other cases the creation of a TransMUX equipment protection group will be rejected.

Error code:

`IDNV – Input, Data Not Valid, invalid slots for worker and protection unit`

Error responses (ES64U protection groups)

Validation rule: ES64U pack already used in other equipment protection group

The creation of an ES64U equipment protection group will be denied if the specified worker or protection pack is already a member of another ES64U equipment protection group.

Error code:

`SNVS – Status, Not in Valid State, circuit pack is member of an equipment protection group`

Validation rule: ES64U switch capacity and unit roles

The creation of the ES64U equipment protection group is rejected, if the worker unit has an ES64U switch capacity of 0 GBit/s or if the protection unit has an ES64U switch capacity greater than 0 GBit/s.

Error code:

`SNVS – Status, Not in Valid State, ES64U switch capacity invalid`

The creation of an ES64U equipment protection group is only allowed, if the worker ES64U pack has *es64ucap* > 0 and the protection pack has *es64ucap* = 0.

Error responses (general)

Validation rule: port on different packs

Each port, being a member within the protection group, is located on different circuit packs.

Error code:

`SNVS – Status, Not in Valid State, ports must be on different circuit packs`

Validation rule: common interface type

Each port being a member within the protection group, shall have the same interface standard i.e. all ports shall either operate in SONET mode or in SDH mode.

Error code:

`SNVS – Status, Not in Valid State, ports have different interface types`

Validation rule: supported port rate

The port rate is supported for the specified protection type

Error code:

`SNVS – Status, Not in Valid State, port rate not supported for this protection type`

Value Range: See value range definition for the attribute *pgrate* in the "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759) output section w.r.t. supported port rates for the different protection types.

Validation rules for value range

See value range definition for the attribute *pgrate* in the "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759) output section w.r.t. supported port rates for the different protection types.

Validation rule: common port rate

Each port being a member within the protection group, shall have the same port rate.

Error code:

`SNVS – Status, Not in Valid State, ports have different port rates`

Validation rule: No test access session

Tributaries related to the specified ports must not be used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: XC pack is (pre-)provisioned

There is at least one XC pack pre-provisioned or provisioned.

Error code: `SNVS – Status, Not in Valid State, XC Pack not provisioned`

Validation rule: unique pgid

The specified *pgid* is not already used by an already existing protection group AID of the same protection type.

Error code:

`SDNC – Status, Data Not Consistent, pgid is already used by another protection group of the same type`

Validation rule: maximum number of 2-fibre and 4-fibre MS SPRING / BLSR protection groups

The number of 2-fibre and/or 4-fibre MS SPRING / BLSR protection groups in the system shall not exceed 32.

Error code:

`SSRE – Status, System Resources Exceeded, allowed number of MS SPRING and BLSR protection groups is exceeded`

This restriction results from the number of K-byte express channels which can be supported by the system. The number of groups actually supported by the system is also limited by factors such as the system capacity, the number of ports per slot, the number of ports per group (2-fibre / 4-fibre / 4-fibre asymmetric groups), and the number of DCC channels supported by the CTL (if automatic ring discovery is used).

Error responses (multiplex section/line protection group)

Validation rule: no timing reference on protection port

Protection port is not assigned to a timing reference.

Error code:

`SNVS – Status, Not in Valid State, protection port is assigned to a line timing reference`

Validation rule: no out-of-service state on protection port

Protection port shall not have the value ’OUTOFSERVICE’ for the attribute *servcond*.

Error code:

`SNVS – Status, Not in Valid State, protection port is out-of-service`

Validation rule: no timing reference on other port of protection unit

No other port of unit related to the protection port is assigned to a timing reference if a .timing reference is already assigned to the intended worker port.

Error code:

`SNVS – Status, Not in Valid State, other port of protection unit is assigned to a line timing reference`

Validation rule: No MSP 1:1 protection for bitrate STM-256

The system shall reject an ENT command for the architecture 1:1 (protype = 1xNOPT), if the port bitrate is STM-256.

Error code:

`IDNC – Input, Data, Not Consistent, This MSP protection type not supported for a port with the indicated rate`

Validation rule: No MSP 1+1 optimized protection for SONET

The system shall reject an ENT command for the architecture 1+1 (protype = 1+1) if it is combined with the operation parameter value ’optimized’ (protocol =1+1_OPTM) and if the port mode is SONET.

Error code:

`IDNC – Input, Data Not Consistent, MSP 1+1_optimized not supported for SONET mode`

Validation rule: No MSP 1:1 protection for SONET

The system shall reject an ENT command for the architecture 1:1 (protype = 1xNOPT), if the port mode is SONET.

Error code:

`IDNC – Input, Data, Not Consistent, No MSP protection supported for SONET mode`

Validation rule: MSP type 1:1 is never non-revertive

The system shall reject an ENT command for the architecture 1:1 (protype = 1xNOPT), if it is combined with the operation mode ’non-revertive’ (rme=disabled).

Error code:

`IDNC – Input, Data, Not Consistent, MSP Protection Type 1x1 is always revertive`

Validation rule: No MSP/APS 1+1 revertive protection for bitrate STM-256/OC-768

The system shall reject an ENT command for the architecture 1+1 (protype = 1+1) and the operation mode 'revertive' (rme=enabled) , if the port bitrate is STM-256 or OC-768.

Error code:

`IDNC – Input, Data Not Consistent, revertive mode not supported for MSP Protection Type 1+1 and a port with the indicated rate`

Validation rule: No MSP/APS 1+1 for bitrate STM-256/OC-768

The system shall reject an ENT command for the architecture 1+1 (protype = 1+1), if the port bitrate is STM-256 or OC-768.

Error code:

`IDNC – Input, Data Not Consistent, This MSP protection type not supported for a port with the indicated rate`

Validation rule: only MSP 1+1 can be optimized

The system shall reject an ENT command for any architecture beside 1+1 (protype = 1+1) if it is combined with the operation parameter value 'optimized'.

Error code:

`IDNC – Input, Data, Not Consistent, Optimized exists only with 1+1 Protection Type`

Validation rule: No MSP 1+1 optimized for bitrate STM-256

The system shall reject an ENT command for the architecture 1+1 (protype = 1+1), if combined with the parameter protocol=OPTM and the port bitrate STM-256.

Error code:

`IDNC – Input, Data, Not Consistent, This MSP protection type not supported for a port with the indicated rate`

Validation rule: No HO/LO cross-connection or LO substructure at protection tribs

The creation of the Protection group is rejected, if HO or LO cross-connections or LO substructure exist on protection tributaries related to the specified ports.

Error code:

`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

Validation rule: No XC loopbacks on protection tribs

There shall exist no cross-connection loopbacks on protection tributaries related to the specified ports.

Error code:
`SNVS – Status, Not in Valid State, Tributary is used in cross-connection loopback`

Validation rule: No input tributary of LO pathprotected cross-connection on the worker port

No tributary on the dedicated worker port of the 1+1 or 1:1 multiplex section/line protection group is used as input tributary of a lower order path-protected cross-connection.

Error code:
`SNVS – Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO crossconnection`

Validation rule ID 69476 disallows any cross-connection at the dedicated protection port. The tributaries at the worker port cannot be used as input tribs in pathprotected LO cross-connections. This is checked in the validation rule ID 69570 above.

Any other cross-connection is allowed at the worker port, i.e.

- HO point-to-point cross-connections
- HO path protected cross-connections
- LO point-to-point cross-connections
- Output of LO path protected cross-connections

Validation rule: MSP type 1+1_optimized is never revertive

The system shall reject an ENT command for the architecture 1+1 (protype = 1+1) if it is combined with the operation parameter value 'optimized' and with the operation mode "'revertive'" (*rme* = enab.).

Error code:
`IDNC – Input, Data, Not Consistent, MSP 1+1_Optimized is always non-revertive`

Validation rule: supported *portClass*

Ports must not be of *portClass* INNI

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: no EDGE protection port for 1+1 MSP

In case of a 1+1 MSP group the specified protection port is not of portClass EDGE

Error code: `SNVS – Status, Not in Valid State, portClass may not be EDGE`

Validation rule: no EDGE protection port for 1:1 MSP

In case of a 1:1 MSP group the specified protection port is not of portClass EDGE.

Error code: `SNVS – Status, Not in Valid State, portClass may not be EDGE`

Validation rule: no UNI protection port for 1+1 MSP

In case of a 1+1 MSP group the specified protection port is not of portClass UNI

Error code: `SNVS – Status, Not in Valid State, portClass may not be UNI`

Validation rule: no UNI protection port for 1:1 MSP

In case of a 1:1 MSP group the specified protection port is not of portClass UNI.

Error code: `SNVS – Status, Not in Valid State, portClass may not be UNI`

Error responses (2-fibre MS-SPRing/BLSR protection group)

Validation rule: unique *rid*

The specified *rid* is not already used by an already existing protection group of type 2F, 4F.

Error code:
`IDNV – Input, Data Not Valid, rid is already used by another protection group`

Validation rule: dcc enabled

If the parameter ard (automatic ring discovery) is not DISABLED, for each side at least one dcc channel (section/dcc-r or line/dcc-m) is enabled.

Error code: `SNVS – Status, Not in Valid State, no dcc channel enabled`

Validation rule: Automatic squelch map calculation only with automatic ring discovery

If the parameter ard is specified with the value DISABLE (automatic ring discovery disabled), then the parameter *asmc* must not be set to ENABLE (default value).

Error code:
`IDNV – Input, Data Not Valid, Automatic Squelch Map Calculation only allowed with Automatic Ring Discovery`

Validation rule: supported *portClass*

Ports must not be of *portClass* INNI

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: supported *portClass*

Ports must not be of *portClass* UNI

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: No HO/LO cross-connection or LO substructure at tributaries

The creation of the protection group is rejected, if HO or LO cross-connections or LO substructure exist on tributaries related to the specified ports.

Error code:
`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

Validation rule: no XC loopbacks on tribs

There shall exist no cross-connection loopbacks on tributaries related to the specified ports.

Error code:
`SNVS – Status, Not in Valid State, Tributary is used in cross-connection loopback`

Error responses (4-fibre MS-SPRing/BLSR protection group)

Validation rule: unique *rid*

The specified *rid* is not already used by an already existing protection group of type 2F, 4F.

Error code:
`IDNV – Input, Data Not Valid, rid is already used by another protection group`

Validation rule: dcc enabled

If the parameter *ard* (automatic ring discovery) is ENABLED, for each worker port at least one dcc channel (section/dcc-r or line/dcc-m) is enabled.

Error code: `SNVS – Status, Not in Valid State, no dcc channel enabled`

Validation rule: dcc enabled on protection port

If the parameter ard (automatic ring discovery) is ENABLED, for each protection port of an asymmetric span at least one dcc channel (section/dcc-r or line/dcc-m) is enabled.

Error code: `SNVS – Status, Not in Valid State, no dcc channel enabled`

The characteristic of an asymmetric ring is that one or more spans have only protection lines, no working lines (asymmetric spans). In these asymmetric spans it must be possible to use the DCC channels of the protection ports.

Automatic Squelch Map Calculation only with Automatic Ring Discovery

If the parameter *ard* is specified with the value DISABLE (automatic ring discovery disabled), then the parameter *asmc* must not be set to ENABLE (default value).

Error code:
`IDNV – Input, Data Not Valid, Automatic Squelch Map Calculation only allowed with Automatic Ring Discovery`

Validation rule: Timing reference on protection port only allowed in asymmetric spans of a ring

The system shall not allow to assign the protection port of a symmetric span of a ring to a timing reference. The assignment of the protection port to a timing reference is only allowed for asymmetric spans.

Error code:

`SNVS – Status, Not in Valid State, protection port is assigned to a line timing reference`

The characteristic of an asymmetric ring is that one or more spans have only protection lines, no working lines (asymmetric spans). In these asymmetric spans it must be possible to use the protection ports for timing.

Validation rule: no out-of-service state on protection port

Protection port shall not have the value ’OUTOFSERVICE’ for the attribute *servcond*.

Error code:

`SNVS – Status, Not in Valid State, protection port is out-of-service`

Validation rule: no timing reference on other port of protection unit

No other port of unit related to the protection port is assigned to a timing reference if a timing reference is already assigned to the intended worker port.

Error code:

`SNVS – Status, Not in Valid State, other port of protection unit is assigned to a line timing reference`

Validation rule: SDH ports for TOP

If *protnmode* equals ’SHORTENED’ or ’TOPEX’ (both are TOP), then all specified member ports shall have the interface standard SDH.

Error code:

`SNVS – Status, Not in Valid State, SONET port not allowed for TOP`

Validation rule: complete side information

The parameters *eastwkg*, *eastprotn*, *westwkg*, *westprotn* must be present in the request.

Error code: `IDNV – Input, Data Not Valid, side information is not complete`

Validation rule: supported *portClass*

Ports must not be of *portClass* INNI

Error code: `SNVS – Status, Not in Valid State, portClass must not be INNI`

Validation rule: supported *portClass*

Ports must not be of *portClass* UNI

Error code: `SNVS – Status, Not in Valid State, portClass must not be UNI`

Validation rule: supported *portClass*

Protection ports must not be of *portClass* EDGE

Error code:

`SNVS – Status, Not in Valid State, portClass of protection port must not be EDGE`

Validation rule: supported *portClass* for TOP

If *protnmode* equals SHORTENED or TOPEX (both are TOP), then none of the specified member ports shall have the *portClass* EDGE.

Error code:

`SNVS – Status, Not in Valid State, portClass must not be EDGE for TOP`

Validation rule: No HO/LO cross-connection or LO substructure at tributaries

The creation of the Protection group is rejected, if HO or LO cross-connections or LO substructure exist on tributaries related to the specified ports.

Error code:

`SNVS – Status, Not in Valid State, cross-connections or substructure still exist`

Validation rule: no XC loopbacks on tribs

There shall exist no cross-connection loopbacks on tributaries related to the specified ports.

Error code:

`SNVS – Status, Not in Valid State, Tributary is used in cross-connection loopback`

Related TL1 messages

- "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)
- "ED-PROTN-GRP: Edit protection group" (p. 3-803)
- "DLT-PROTN-GRP: Delete protection group" (p. 3-233)
- "OPR-PROTNSW: Operate protection switch" (p. 3-2057)
- "RLS-PROTNSW: Release protection switch" (p. 3-2203)
- "ED-MAP-RING: Edit map ring" (p. 3-436)
- "SET-MAP-RING: Set map ring" (p. 3-3142).

ENT-PROTN-GRP notification

Output format

The following syntax is used for 1+1 equipment, 1+1 multiplex section/line, 2-/4-fibre MS-SPRing/BLSR protection group.

General Syntax: **`ENT-PROTN-GRP`**:*aid*:*protype:spec_block*;

Path Protection Group Syntax: **`ENT-PROTN-GRP`**:*aid*:*protype:spec_block {outtribaid, CML: cml_block}+*;

CML Block Distribution on 2 Notifications: If the number of CML blocks to be sent in an ENT-PROTN-GRP Notification exceeds the capabilities of the system, then the CML blocks is sent in 2 separate ENT-PROTN-GRP Notifications. The sets of CML blocks contained in the 2 ENT-PROTN-GRP Notifications is disjoint.

Output parameters

Command parameters (1+1 equipment protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 1+1 protection equipment group.
   Type: AID
   Range: SwitchAndTmgProtGrpAID for switching protection group instance; CtlProtGroupAID for controller protection group instance; EP155ProtGroupAID for EP155 protection group instance, TMUXProtGroupAID for TransMUX groups, ES64UProtGroupAID for ES64U groups.
2. **`protype`** – Protection Type
   The parameter specifies the type of the protection group for retrieving the information.
   Type: Alphanumeric value set.

   Range:

   - EQPTSWFBR (if *aid* is SwitchAndTmgProtGrpAID)
   - EQPTCTL (if *aid* is CtlProtGroupAID)
   - EQPTEP155 (if *aid* is EP155ProtGroupAID)
   - EQPTEP51 (if *aid* is EP51ProtGroupAID)
   - EQPTLOXC (if *aid* is LOXCProtGroupAID)
   - EQPTTMUX (if *aid* is TMUXProtGroupAID)
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

Command parameters (multiplex section/line protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later).
   Type: AID
   Range: OnePlusOne_ProtGrpAID or NByOne_ProtGrpAID
2. **`protype`** – Protection Type
   The parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1x***N***OPT (only R3.0 and later).
   In mind the ***N*** in "1x***N***OPT" is considered to be always equal "1".
   Type: Alphanumeric value set.
   Range: 1+1 (default) or 1xNOPT.

Command parameters (2-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 2 fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: TwoFibre_ProtGrpAID
   Default: No default
2. **`protype`** – Protection Type
   The parameter specifies the type of the protection group. The value for the 2-fibre MS-SPRing/BLSR protection group is 2F.
   Type: Alphanumeric value set.
   Range: 2F.

Command parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 4 fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: FourFibre_ProtGrpAID
2. **`protype`** – Protection Type
   The parameter specifies the type of the protection group. The value for the 4-fibre MS-SPRing/BLSR protection group is 4F.
   Type: Alphanumeric value set.
   Range: 4F.

Command parameters (Path protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the path protection group.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.
2. **`protype`** – Protection Type
   The parameter specifies the type of the protection group. The value for the path protection group is PATHDRI.
   Type: Alphanumeric value set.
   Range: PATHDRI
3. **`outtribaid`** – Output Tributary Access Identifier
   This parameter specifies the output tributary AID for a constituent signal within the protection group. One value is returned for each constituent signal together with the corresponding CML (Constituent Member List) block.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.

Spec block parameters (1+1 equipment protection group)

1. **`protn`** – Protection Unit
   The parameter specifies the protection circuit pack AID being a member of the 1+1 equipment protection group.
   Type: AID
   Default: For switching protection group instance: CpAID related to slot 10. For controller protection group instance: CpAID related to slot 31. For EP155 protection group instance: – If no cross-connections are assigned to one of the two EP155 cards, the user can choose it by the **`ENT-PROT-GRP`** command Default protection unit is the right card. – If cross-connections are assigned to one of the cards, the other card must become the protection unit. This means that the creation (ENT-PROTN-GRP) is rejected, if the user selects the card with cross-connections as protection card.
2. **`wkg`** – Worker Unit
   This parameter specifies the worker circuit pack AID being a member of the 1+1 equipment protection group.
   Type: AID
   Range: CpAID
   Default: For switching protection group instance: CpAID related to slot 9. For controller protection group instance: CpAID related to slot 11. For EP155 protection group instance: – If no cross-connections are assigned to one of the two

EP155 cards, the user can choose it by the **ENT-PROT-GRP** command Default worker unit is the left card. – If cross-connections are assigned to one of the cards, this card must become the worker unit. This means that the creation (ENT-PROTN-GRP) is rejected, if the user selects the card with no cross-connections as worker card.

Spec block parameters (multiplex section/line protection group)

1. **apspn** – MSP-APS ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MSP-APS (Multiplex Section Protection – Automatic protection switching).
   Type: Alphanumeric string
   Range: 1-24 characters including character '-'.
   Default: DEFAULT
2. **dcclmsprotmd** – DCC Line/MS Protection Mode
   The parameter specifies, whether the line/MS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if DCC channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set 3;
   Range: ENABLE, DISABLE
   Default: DISABLE
3. **dccsrsprotmd** – DCC Sect/RS Protection Mode
   The parameter specifies, whether the section/RS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set
   Range: ENABLE, DISABLE
   Default: DISABLE
4. **protn** – Protection Unit
   This parameter specifies the protection port AID being a member of the line protection/ MS protection group.
   Type: AID
   Range: SPortAID

5. **wkg** – Worker Unit
   This parameter specifies the worker port AID being a member of the 1+1 line/MS protection group.
   Type: AID
   Range: SPortAID
6. **wtr** – Wait To Restore Time
   The parameter specifies the “wait to restore” time used by the protection state machine operating in revertive mode. In case of MSP 1+1 optimized, the same parameter is used for the “wait to rename” time.
   Type: numeric string [minute]
   Range: 0 – 60
   Default: 5.

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

1. **ard** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value manually with ED-MAP-RING and supply the ring map information to each node on the ring for the BLSR protocol to operate. It is necessary to activate this user-provisioned information with SET-MAP-RING. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
   Type: Alphanumeric string.
   Range: ENABLE, DISABLE.
   Default: ENABLE.
2. **asmc** – Automatic Squelch Map Calculation
   If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
   Type: Alphanumeric string
   Range: ENABLE, DISABLE (default)
3. **blsrpn** – MS-SPRing-BLSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
   Type: Alphanumeric string
   Range: 1-24 characters including the character “-”.
   Default: DEFAULT
4. **east** – East Port
   This parameter specifies the port being a member in the role of the east port for a 2-fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: SPortAID

5. **rcamode** – Ring Circuit Alarm Mode
   This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
6. **rid** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
7. **west** – West Port
   This parameter specifies the port being a member in the role of the west port for a 2-fibre MS-SPRing/BLSR protection group.
   Type: AID.
   Range: SPortAID.
   Default: None.
8. **wtr** – Wait To Restore Time
   This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
   Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
   Type: numeric string [minute].
   Range: 0 – 12 in 1 minute increments and 99.
   Default: 5.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

1. **ard** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value and supply the ring map information to each node on the ring for the BLSR protocol to operate. Until all

nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.

2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: Alphanumeric string
Range: ENABLE, DISABLE (default)
3. **`blsrpn`** – MS-SPRing-BLSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
Type: Alphanumeric string.
Range: 1-24 characters including the character "-".
Default: DEFAULT.
4. **`eastprotn`** – East Protection Port
This parameter shall specifies the port being a member in the role of the east protection port for a 4-fibre MS-SPRing/BLSR protection group.
Type: AID
Range: SPortAID
5. **`eastwkg`** – East Worker Port
This parameter specifies the port being a member in the role of the east worker port for a 4-fibre MS-SPRing/BLSR protection group.
Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID.
Range: In case of an asymmetric ring: NOTUSED. In all other cases: SPortAID.
6. **`protnmode`** – Protection Switch Mode
The parameter specifies the protection switch mode. The value LOOPBACK is used for the classical ring application. The value SHORTENED is used for transmission delay sensitive applications. The value TOPEX (only R3.0 - R3.2 and R8.0.1 and later) is used for the so called transoceanic application, which allows extra traffic, improves restorability and improves delay as well as in SHORTENED mode.
In case of a ring switch … LOOPBACK performs the bridge and switch function in that nodes which are most close to the failed span. SHORTENED performs the bridge function at the add node repectively the switch function at the drop node of each ring circuit. It is an application only similar to the transoceanic protocol.
TOPEX allows, compared to SHORTENED mode, additionally the traffic restoration of not affected spans. It is the transoceanic protocol application.
Consistent Protection Switch Mode Setting: The Protection Switch Mode must be set consistently on all nodes in the ring. E.g. for all UNITE NEs in a TOP+EX ring the parameter protnmode must be set to TOPEX. In a ring with inconsistent

Protection Switch Mode, there will be no protection switching. The UNITE NEs will raise an Inconsistent Ring Protection Mode alarm.
Type: Alphanumeric value set.
Range: LOOPBACK, SHORTENED, TOPEX.
Default: LOOPBACK.

7. **`rcamode`** – Ring Circuit Alarm Mode
This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If rcamode is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE

8. **`rid`** – Ring Identification
This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE

9. **`swtr`** – Span Wait To Restore Time
The parameter specifies the wait to restore time for span switches used by the protection state machine operating in revertive mode. The protection shall operate in revertive mode if the wait-to-restore time is in the range 0-12. The protection shall operate in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

10. **`westprotn`** – West Protection Port
This parameter specifies the port being a member in the role of the west protection port for a 4-fibre MS-SPRing/BLSR protection group.
Type: AID
Range: SPortAID

11. **`westwkg`** – West Worker Port
    This parameter specifies the port being a member in the role of the west worker port for a 4-fibre MS-SPRing/BLSR protection group
    Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID
    Range: In case of an asymmetric ring: NOTUSED; in all other cases: SPortAID.
12. **`wtr`** – Wait To Restore Time
    This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
    Note: If wtr time is set to infinite, the switch request state is WTR until a higher request occurs.
    Type: numeric string [minute].
    Range: 0 – 12 in 1 minute increments and 99.
    Default: 5.

Spec block parameters (path protection group)

1. **`cst`** – Constituent Signal Type
   In cases of mixed usage of SONET and SDH terms in provisioning of cross-connection and provisioning of protection group no conversion will happen. The equivalent term may be shown instead. For example: Assume the protection group port rate – identified by the modifier – is entered as a SDH term. If before the underlying cross-connection rate has been entered as a SONET term the parameter cst will nevertheless show the SONET term according ID 24284 (e.g. STS48C).
   This parameter specifies the rate of the path protection group.
   Type: Alphanumeric value set.
   Range: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C.
2. **`dripathpn`** – SNCP-UPSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type SNCP-UPSR.
   Type: Alphanumeric string.
   Range: 1-24 characters including the character “-”.
   Default: DEFAULT.
3. **`htmr`** – Hold-Off Time
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms].
   Range: 0 .. 99 ms in 1 ms steps and 100 .. 10000 ms in 100 ms steps.

4. **`ppgname`** – Path protection group Name
   This parameter specifies the path protection group name. Storage and retrieval of the *ppgname* is case sensitive, but actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
   Type: text string.
   Range: 0 to 24 characters.
   Default: empty string
5. **`protn`** – Protection Tributary
   This parameter specifies the protection input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
6. **`rme`** – Protection Reversion
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group.
   The value ENABLE is used if the protection operates in revertive mode.
   The value DISABLE is used if the protection operates in non-revertive mode.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
7. **`snctype`** – SNC Protection Type
   This parameter specifies the type of the path protection w.r.t. switch criteria used by the protection state machine.
   The value SNCI (Inherent) is used if server signal fail is used as switch criteria.
   The value SNCN (Non-Instrusive) is used if server signal fail and defects supervised by a non-intrusive path monitor are used as switch criteria
   The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
   Type: Alphanumeric value set.
   Range: SNCI, SNCN, NA.
8. **`wkg`** – Worker Tributary
   This parameter specifies the worker input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
   The usage of wild cards is not allowed for any part of the AID.
9. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
   Type: numeric string [minute].
   Range: 0 – 60.

CML block parameters (path protection group)

1. **`cst`** – Constituent Signal Type
   In cases of mixed usage of SONET and SDH terms in provisioning of cross-connection and provisioning of protection group no conversion will happen. The equivalent term may be shown instead. For example: Assume the protection group port rate – identified by the modifier – is entered as a SDH term. If before the underlying cross-connection rate has been entered as a SONET term the parameter cst will nevertheless show the SONET term according ID 24284 (e.g. STS48C).
   This parameter specifies the rate of the constituent signal within the path protection group.
   Type: Alphanumeric value set.
   Range: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C
   Default: The default value is derived from the tributary input signal rate list in case of fixed mode.

Example output

1+1 Equipment Protection Group

The following example reports the creation of an 1+1 equipment protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-PROTN-GRP:1-1-estgrp:EQPTSWFBR:wkg=1-1-#-#-xcw-
      cp,protn=1-1-#-#-xcp-cp″
;
```

Multiplex section/line protection group 1+1

The following example reports the creation of an 1+1 multiplex section/line protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-PROTN-GRP:1-1-o01:1+1:wkg=1-1-#-#-13-1,protn=1-1
      -#-#-14-1,wtr=5,apspn=DEFAULT,
      dcclmsprotmd=DISABLE,dccsrsprotmd=DISABLE,protocol=1+1_UNI,rme=DISABLE″
;
```

2 fibre MS-SPRing/BLSR protection group

The following example reports the creation of a 2 fibre MS-SPRing/BLSR protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-PROTN-GRP:1-1-t001:2F:east=1-1-#-#-13-1,west=1-1
      -#-#-14-1,rid=\″denver\″,wtr=5,blsrpn=DEFAULT″
;
```

4 fibre MS-SPRing/BLSR protection group

The following example reports the creation of a 4 fibre MS-SPRing/BLSR protection group:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-PROTN-GRP:1-1-f001:4F:eastwkg=1-1-#-#-13-
      1,eastprotn=1-1-#-#-15-1,westwkg=1-1-#-#-14-1,westwkg=1-1-#-#-16-
      1,rid=\″denver\″,protnmode=LOOPBACK,swtr=5,wtr=5,blsrpn=DEFAULT″
;
```

Path protection group

The following example reports the creation of a path protection group:

```
LT-UNITE 05-05-31 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-PROTN-GRP:1-1-#-#-10-1-25:PATHDRI:cst=STS1,wkg=1
      -1-#-#-08-1-25,protn=1-1-#-#-06-1-
      1,ppgname=\″group018\″,htmr=100,wtr=5,dripathp=path2,rme=DISABLE,snctype=NA″
   ″1-1-#-#-10-1-25,CML:cst=STS1″
;
```

ENT-PTF notification

Output format

Syntax

`ENT-PTF`-*modifier:aid::spec_block;*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The access identifier specifies the PTF tributary whose attributes are reported.
   Type: AID
   Range:
   - PTNumAID for modifier: STS, HOSTS1, HOVC4, ALL
   - TMuxAPTNumAID for modifier: TMASTS1
   - TMuxSPTNumAID for modifier: HOSTS1, TMSSTS1

   as defined in AID structure with exception of value **`all`**.
2. **`modifier`**
   The modifier specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range:
   - STS for DS3 card
   - HOSTS1 for LOXC card
   - HOVC4 for LOXC card
   - TMASTS1 for TransMUX asynchronous port
   - TMSSTS1 for TransMUX synchronous port

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   The parameter *brstintvl* specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST, then *brstintvl* has no affect to NE behavior.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *brstintvl* is not supported for the following modifiers:
   - HOSTS1
   - HOVC4

- TMASTS1
- TMSSTS1

Option: If the parameter *brstintvl* is not supported, then the parameter is omitted.
Type: Integer.
Range: 2, 3, … 10 [seconds]
Default: 7 [seconds].

2. **`brstthr`** – Bursty Error Threshold
The parameter *brstthr* specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.
The parameter *brstthr* is not supported for the following modifiers:
   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Option: If the parameter *brstthr* is not supported, then the parameter is omitted.
Type: Integer.
Range: 5, 10, 15, … 50 [%].
Default: 30.

3. **`crslpbkstat`** – Cross-connect Loopback Status
The parameter *crslpbkstat* reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).
The parameter *crslpbkstat* is not supported for the following modifiers:
   - HOSTS1
   - HOVC4
   - TMSSTS1

   Option: If the parameter *crslpbkstat* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: YES, for enabled loopback; NO, for disabled loopback

4. **`degthr`** – Signal Degrade Threshold
This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to

RTRV-PTF Output parameters.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter *degthr* is not supported for the following modifiers:

- HOSTS1
- HOVC4
- TMSSTS1

Type: Integer.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

5. **dexcthr** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of sdsfmode refer to RTRV-PTF Output parameters.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *dexcthr* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMSSTS1

   Type: Integer.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.

6. **exphptrc** – Expected HP trace identifier
This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if ptrcrfmt is set to 16.

   Limitation: The parameter is not supported for modifiers:

   - TMSSTS1

   Type: Hexadecimal number without a preceding "H".
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000

7. **`feppm`** – Far End Path PM Enable
   The parameter *feppm* specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
   This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
   Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the parameters for far-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter is not supported for the following modifiers:

   - TMSSTS1

   Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE (default).

8. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion
   The parameter *hptimais* shall specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

   The parameter is not supported for the following modifiers:

   - TMSSTS1

   Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE.

   Default: The default value depends on interface standard specified by *ptfstd* for the PTF tributary. DISABLE, if *ptfstd* = SONET

9. **`hptimmon`** – HP Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).

   The parameter is not supported for the following modifiers:

   - TMSSTS1

   Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE (default).

10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter specifies the input LO signal structure.

The parameter *inputsig* is only supported for the following modifiers:

- HOVC4
- HOSTS1
- TMSSTS1

Option: If the parameter *inputsig* is not supported, then the parameter is omitted.
Type: numeric value set
Range: NX-NX-… (N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
The possible values (SONET) are: 2811
Restricted Range with AU3/AU4 Conversion cross-connections:
If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *inputsig* is restricted to: 33.
Restricted Range with TransMUX:
If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.

Default:

- 6312 for SDH
- 2811 for SONET

11. **neppm** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the possibility to enable near-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1

Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE.

Default: DISABLE.

12. **pmhpfeses** – PM HP Far End SES Threshold
This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *ptfstd* for the PTF tributary. If *ptfstd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.

This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the possibility to enable far-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1

Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

13. **`pmhpneses`** – PM HP Near End SES Threshold
This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *ptfstd* for the tributary. If *ptfstd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1

Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

14. **`ptca`** – Path TCA Profile Pointer
This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

TCA profile: The TCA profile type depends on the type and standard of the PTF, which is specified with the parameter *modifier* or *aidtype*:

- PATH if *modifier* = TMASTS1

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1
- HOVC4

Type: Alphanumeric value set.
Range: 1 … 24 characters.
Default: DEFAULT0

15. **`ptfstd`** – Path Termination Standard
    Standard of the tributary with Path Termination Function.
    This is currently a read-only parameter.
    Type: Alphanumeric value set.
    Range:
    - SDH (only if *modifier* = HOVC4)
    - SONET (only if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1)

    Default:
    - SDH (if if *modifier* = HOVC4)
    - SONET (if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1).
16. **`ptpn`** – Path Terminating ASA Profile Name
    This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
    Range: 1 … 24 characters
    Default: DEFAULT
17. **`ptrcrfmt`** – Path Trace Read Format
    The parameter *ptrcrfmt* specifies the format of the path trace (J1 byte) to be received (sink).
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Numeric string.
    Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.
18. **`ptrcwfmt`** – Path Trace Write Format
    The parameter ptrcwfmt specifies the format of the path trace (J1 byte) to be sent (source).
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Numeric string.
    Range:
    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

    Default: 2.

19. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    This parameter specifies the determination mode of degraded defect for transmission.
    Limitation for PTF related to modifiers: Near-end performance monitoring is not available for modifier STS. The parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1
    - HOVC4
    - HOSTS1

    Option: If the parameter *sdsfmode* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: BURST, POISSON
    Default: The default value depends on interface standard specified by *ptfstd* for the tributary. POISSON, if *ptfstd* = SONET.

20. **`substr`** – Substructured
    The parameter *substr* indicates if the PTF is substructured.
    Additional description for LOXC
    This parameter indicates if the PTF is configured for AU3/AU4 conversion cross-connections.
    The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created. A PTF, which is configured for AU3/AU4 conversion cannot be used to set up HO cross-connections.
    The parameter *substr* is only supported, if *modifier* = STS.
    If the parameter *substr* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.

    Range:

    - CNV: In use for HO side of a conversion cross-connection
    - NO: Not used for conversion cross-connections

    Default: NO.

21. **`tmonmd`** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Type: Alphanumeric value set.
    Range: MON, NMON (default).

22. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Hexadecimal number without a preceding “H”.
    Range: the range depends on the parameter ptrcwfmt specified in RTRV-PTF Output parameters
    - length: 32 hexadecimal digits, if *ptrcwfmt*=16; 128 hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt*=16; 64 byte string if *ptrcwfmt* = 64

    Default: 00. This implies also that *ptrcwfmt* is set to 2.
23. **`tslmmon`** – Trail Signal Label Mismatch Monitor
    The parameter *tslmmon* specifies the mode of the TSL mismatch detection.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Type: Alphanumeric value set.
    Range: DISABLE, ENABLE.
    Default: ENABLE.
24. **`unequiposig`** – Unequipped Output Signal List
    The parameter *unequiposig* specifies the output LO signal structure, independent whether the signal is connected or not. The parameter *unequiposig* is only supported for modifiers HOVC4. HOSTS1. and TMSSTS1.
    Option: If the parameter *unequiposig* is not supported, then the parameter is omitted.
    Type: numeric value set
    Range: NX-NX-…(N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
    The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
    The possible values (SONET) are: 2811
    Restricted Range with AU3/AU4 Conversion cross-connections:
    If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *unequiposig* is restricted to: 33.
    Restricted Range with TransMUX:
    If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.
    Default:
    - 6312 for SDH
    - 2811 for SONET

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:ENT-PTF-STS:1-1-#-#-21-4-1::
      brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-
      3,hptimais=DISABLE,exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,ptpn=DEF
;
```

ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers

A routing controller (RC) in this list is considered to be a (logical) adjacent RC and routing information is exchanged with this RC.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S3

Priority: 1

Abortable: No.

Input format

Command syntax: **`ENT-RC-NEIGHBOR`**:*tid*::*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **`remrcid`** – Remote Routing Controller Identifier
   This parameter is used to indicate the Id of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
2. **`remrcscn`** – Remote Routing Controller SCN IP Address
   This parameter is used to indicate the IP address of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.

3. **remrcraid** – Remote Routing Controller Routing Aread ID
   This parameter is used to indicate the routing area ID of the level 0 area of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   0.0.0.0 is not allowed.
4. **remrcOspfFlooding** – Remote Routing Controller OSPF Flooding
   This parameter indicates whether OSPF flooding is enabled to the remote RC or not.
   Type: Alphanumeric value set.
   Range: DISABLE or ENABLE.

Output format

If the **ENT-RC-NEIGHBOR** command is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″[rate]:thru[,lpaid]:spec_block″
   ...
   ″[rate]:thru[,lpaid]:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the “RTRV-HDR: Retrieve header” (p. 3-2417) command.

Detailed behavior description

The **ENT-RC-NEIGHBOR** command adds a routing controller to the list of neighboring routing controllers. The RC neighboring list is used to create level 1 adjacencies for DDRP / OSPF-TE routing.

Example input/output

The following example shows the result of a ENT-RC-NEIGHBOR command:

```
RTRV-MAP-NEIGHBOR:LT-UNITE::123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=SECTION″
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=LINE″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ENT-RC-NEIGHBOR** command.

Validation rule: RC must be created

If the RC is not created (via ACT-L1ROUTING), then the command is denied.

Error code: `IDNV – Input, Data Not Valid, RC node not created`

Validation rule: Neighboring RC must differ from own RC

The command is denied if the neighboring RC ID identified by *remrcid* and/or the neighboring IP address identified by *remrcscn* is identical with the own RC id respectively own RC IP SCN address.

Error code:

`IDNV – Input, Data Not Valid, neighboring RC ID/IP address used by own RC`

Validation rule: RC must not be in Neighboring List

The command is denied if the routing controller is already in the neighboring RC list, i.e. the command is denied if either *remrcid*, *remrcscn*, or *remrcraid* are already in the neighboring RC list.

Error code: `IDNV – Input, Data Not Valid, RC neighbor already provisioned`

Validation rule: Maximum Number of Neighboring RCs

The command is denied if the neighboring RC list already contains 64 entries.

Error code:

`SSRE – Status, System Resources Exceeded, allowed limit of neighboring RCs exceeded`

Related TL1 messages

- "RTRV-RC-NEIGHBOR: Retrieve all neighboring routing controllers" (p. 3-2816)
- "DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers" (p. 3-247)

ENT-RC-NEIGHBOR notification

Output format

The **ENT-RC-NEIGHBOR** notification has the following specific syntax:

ENT-RC-NEIGHBOR:::*spec_block*

Output parameters

Spec block parameters

1. **remrcid** – Remote Routing Controller Identifier
This parameter is used to indicate the Id of the remote routing controller.
Type: Four dot separated numeric strings.
Range: Each numeric string: 0 .. 255.
2. **remrcscn** – Remote Routing Controller SCN IP Address
This parameter is used to indicate the IP address of the remote routing controller.
Type: Four dot separated numeric strings.
Range: Each numeric string: 0 .. 255.
3. **remrcraid** – Remote Routing Controller Routing Aread ID
This parameter is used to indicate the routing area ID of the level 0 area of a neighboring route controller.
Type: Four dot separated numeric strings.
Range: Each numeric string: 0 .. 255.
0.0.0.0 is not allowed.
4. **remrcOspfFlooding** – Remote Routing Controller OSPF Flooding
This parameter indicates whether OSPF flooding is enabled to the remote RC or not.
Type: Alphanumeric value set.
Range: DISABLE or ENABLE.

Example output

```
RTRV-MAP-NEIGHBOR:LT-UNITE::123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=SECTION″
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=LINE″
;
```

ENT-ROLL: Enter roll

The **ENT-ROLL**-*modifier* command can be initiated by a user to roll the input of a cross-connection from a given tributary to another tributary while leaving the output unchanged. The **ENT-ROLL** command can be modelled by a DLT-CRS (deleting the existing cross-connection) followed by a ENT-CRS (establishing the new cross-connection according to the specified new input-tributary). This also holds for path-protected cross-connections for which the input of one of the legs can be rolled via the **ENT-ROLL** command, while leading the output of the path selector unchanged.

The tributary to which the roll cross-connection is to be established does not need to be free of alarms/failure conditions – no check in advance will be made. The cross-connection will thus be established also in case of AIS-P, LOP-P, UNEQ etc. as is the case with a normal ENT-CRS command as well.

The *modifier* in the **ENT-ROLL** command indicates the cross-connection rate of the tributary for which the command is applied to. There is no difference if the rate is specified as a SONET rate (STS1, STS3,..) or as SDH-rate (VC3, VC4,..) for the deleting of the old and establishing of the new cross-connection i.e. the ENT-ROLL-<*sonet-rate*> yields to the same physical and logical behavior as ENT-ROLL-<*sdh-rate*>.

No check is made if the *modifier* matches the interface-standard of the port i.e. if the port is provisioned as SONET (resp. SDH) and the rate is a SONET (resp. SDH) rate.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 3

Abortable: No

Input format

Command syntax: **ENT-ROLL**-*modifier*:*tid*:*in_aid*,*out_aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection on which the **ENT-ROLL** command acts on.
   For detailed description please refer to the “RTRV-CRS: Retrieve cross-connection” (p. 3-2306) command.
   Type: Alphanumeric value set.

Range (SDH):

- LOVC12
- LOVC3
- VC3
- VC4
- VC44C
- VC416C
- VC464C

Range (SONET):

- LOVT1
- STS1
- STS3
- STS12
- STS48
- STS192
- AU3TU3
- TU3AU3.

No use of the **all** value is allowed.

2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **in_aid** – Input Access Identifier
   This parameter specifies the source of the cross-connection.
   For a detailed description please refer to the in_aid output parameter in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
   Type: GenTribAID or GenLOTribAID.
   Range: No use of the **all** or **usall** or **vall** values within the in_*aid* are allowed.
4. **out_aid** – Output Access Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   For a detailed description please refer to the out_aid output parameter in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
   Type: GenTribAID or .GenLOTribAID
   Range: No use of the **all** or **usall** or **vall** values within the *out_aid* is allowed.
5. **ctag** – Correlation Tag
   See definition of *ctag* in “RTRV-HDR: Retrieve header” (p. 3-2417).

Spec block parameters

1. **`newinaid`** – New Input Access Identifier
   This parameter specifies the source of the new cross-connection to be established after the existing one is deleted.
   Type: GenTribAID or GenLOTribAID.
   STS1NumAID are used for high order tributaries on e.g. OP***n*** or EP***n*** packs, VCGNumAID are used for high order tributaries on e.g. GE***n*** or OPT2G5 packs, GenLOTribAID are used for low order tributaries on I/O packs e.g. OP***n*** or EP155.
   Range: n. a.
   No use of the **`all`**, **`vall`** or **`usall`** values within the *newinaid* is allowed.

2. **`newloca`** – New Location-a
   This parameter specifies the new source TID if the *newinaid* is in a BLSR protection group.
   Additional information are given for this parameter in the RTRV-CRS section with respect to: figure illustrating BLSR connections, inter-ring cross-connections with collapsed nodes and extra traffic handling.
   loca (location-a) specifies the TID of the node at which service is added to the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections, if the Automatic Squelch Map Calculation function is disabled.. The parameter loca can be specified for BLSR THROUGH, INTER-BLSR and DROP connections while the Automatic Squelch Map Calculation function is enabled to do a User Override.
   The *loca* parameter value has to be omitted for ADD, non-BLSR connections. A 2way cross-connection is considered to be a DROP if the out_aid is a non BLSR high-order tributary.
   For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring node then *loca* is the TID of the node where the related HO circuit is first added to the ring. Refer to the Fig. in the loca description of this section for further explanation.
   At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
   Additional information are given for this parameter in the RTRV-CRS section with respect to:
   - figure illustrating BLSR connections,
   - inter-ring cross-connections with collapsed nodes,
   - extra traffic handling.

   Type: TID
   Range: 1 to 20 characters. Needs to be the same as defined in RTRV-HDR.

3. **`newppgname`** – New Path protection group Name
   This parameter specifies the name of the new path protection group. This parameter is applicable if a leg of a path-protected cross-connection is rolled – the parameter in this case is optional and if not entered the default is used. For non path-protected cross-connection this parameter does not apply and is omitted.
   Note: Storage and retrieval of the ppgname is case sensitive, but the actual use of the ppgname to determine the corresponding protection group is case insenitive.
   This parameter is applicable to path-protected high order cross-connections only and is omitted for non path-protected cross-connections and any low order cross-connection (independent whether unprotected or protected).
   Default: NULL

4. **`xcappl`** – cross-connection Application
   This parameteris a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies.
   The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.
   The numeric values are assigned to approriate textstrings (see Range below) and the Manager should display these value in case of a RTRV-CRS.

   As a possible assignment proposal the following may serve:

   - 0) Unknown
   - 1) 1-Way Point-to-Point
   - 2) 1-Way Path-Protected
   - 3) 1-Way Adjunct Path-Protected
   - 4) 1-Way Conversion Cross-Connection
   - 10) 2-Way Point-to-Point
   - 11) 2-Way Conversion Cross-Connection
   - 20) UPSR (or SNCP Ring) Add, Drop
   - 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
   - 22) UPSR (or SNCP) Drop, 1-Way Broadcast
   - 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
   - 30) Logical Ring (or SNCP) Add, Drop
   - 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
   - 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
   - 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
   - 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)

- 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
- 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

Type: numerical value set
Range: 0 (default), 1, 2.., 255

5. **xcnum** – cross-connection Number
This parameter identifies each cross-connection leg in a specific compound cross-connection topology. This parameter is optional – if not entered the default is used.
Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in Ring Interworking scenarios)
Type: numeric value set (12 digit)
Two digits for the bay, one digit for the shelf, two digits for the slot, two digit for the port, three digits for the high-order tributary, and two digits for the low-order tributary.
If a high-order tributary is addressed then the last two digits are set to 00.
Range: 000000000000 (default).. 999999999999

Output format

If the ENT-Roll command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

Behavior

Because the ENT-ROLL can be understood as a sequence consisting of a DLT-CRS operation followed by an ENT-CRS operation there does not exists a (separate) ENT-ROLL notification. Instead a DLT-CRS notification is sent, followed by an ENT-CRS notification. Furthermore additionally an ED-CRS notification is triggered by the **ENT-ROLL** command, when the *repleg* status is altered. Refer also to the section “DLT-CRS: Delete cross-connection” (p. 3-145) and “ENT-CRS: Enter cross-connection” (p. 3-1464) behavior.

Transaction behavior

For the **ENT-ROLL** command a validation is performed before the COMPLD is returned. This means it has to be checked if the DLT-CRS (using the validation rules as stated in the DLT-CRS section) as well as the following ENT-CRS (by using the validation rules as stated in the ENT-CRS section) can be performed successfully, BEFORE the command is executed. With this approach it is guaranteed that the command either is rejected or succeeds as a whole by executing the deletion- AND the entering-operation.

In case of low-order cross-connections a ENT-ROLL command may lead to rolling multiple cross-connections sharing the specified input tributary. In this case all all validations for the deletion of the old cross-connections and the creation of the new cross-connections have to be successfully passed before sending the COMPLD indication.

Updated affected entities: 1-way, 2-way cross-connection

Refer to the section *Detailed behavior description* of the “ENT-CRS: Enter cross-connection” (p. 3-1464) and “DLT-CRS: Delete cross-connection” (p. 3-145) commands for the affected entities.

Updated affected entities: path-protected cross-connection

If one leg of path-protected cross-connection is rolled, the following updates takes place:

1. The path-protected cross-connection instance is deleted and this is notified via one DLT-CRS notification.
2. Two legs composing the path-protected XCs are created and this is notified via ENT-CRS notification. The initial values for the cross-connection legs are derived from:
   - the data of the deleted leg not specified in the roll request
   - the data as specified in the roll request; i.e. for the rolled leg the attributes: *loca*, *xcappl*, *xcnum*.
3. The attribute *wkg* (if worker leg is rolled) or *protn* (if protection leg is rolled) are updated (gets the value of the new input trib) for the associated path protection group. If in addition a new value for *ppgname* is specified, then the corresponding protection group is updated accordingly. These updates are reported by means of a ED-PROTN-GRP notification.

For details on protection group and the related notifications please refer to the protection group section.

Deletion of the leg non-affected by the roll command is needed, since also the *in_aid2* attribute for this leg is changed. Attribute *in_aid2* is considered as part of the key/identifier for the leg itself, and therefore the deletion and creation sequence is the only approach in order to notify the performed DB changes.

Updated affected entities: low-order path-protected cross-connection in Emulated Selector Bridges

The rolling of a path protected cross-connection being part of a low-order emulated selector bridge construct is notified by the system towards the management system interface by two removal notification for the both old path-protected cross-connections sharing the same input tributary followed by two times two establishment notifications for each of the new cross-connection legs established (two per path-protected cross-connection).

If one leg of pathprotected crossconnection being part of a low-order emulated selector bridge construct is rolled, the following updates shall take place:

1. The two pathprotected cross-connection instances sharing the input trib to be rolled are deleted and this is notified via two DLT-CRS notification.
2. Creation of path-protected cross-connections:
   - Two legs composing the first path-protected XC are created and this is notified via ENT-CRS notification.
   - Two legs composing the path-protected XC are created and this is notified via ENT-CRS notification.
   - The initial values for the cross-connection legs are drived from: the data of the deleted leg not specified in the roll request; the data as specified in the roll request; i.e. for the rolled leg the attributes: *loca*, *xcappl*, *xcnum*.
3. The attribute *wkg* (if worker leg is rolled) or *protn* (if protection leg is rolled) are updated (gets the value of the new input trib) for the associated path protection group. If in addition a new value for *ppgname* is specified, then the corresponding protection group is updated accordingly. These updates are reported by means of a ED-PROTN-GRP notification.

For details on protection group and the related notifications please refer to the protection group section.

Deletion of the legs non-affected by the roll command is needed, since also the *in_aid2* attribute for this leg is changed. Attribute *in_aid2* is considered as part of the key/identifier for the leg itself, and therefore the deletion and creation sequence is the only approach in order to notify the performed DB changes.

Updated affected entities: bridged lower order cross-connections

If a lower cross-connection, which shares its input tributary with other lower order cross-connections (i.e. bridged cross-connection), is rolled to a new input tributary, then all cross-connections using the same input tributary is rolled to the new input tributary.

This results in the following updates and notifications:

1. The cross-connection instances sharing the input trib to be rolled are deleted and this is notified via DLT-CRS notifications.
2. New cross-connections using the new input trib are created and this is notified via ENT-CRS notifications.

Updated affected entities: bridged AU3/AU4 conversion cross-connections

If an AU3/AU4 conversion cross-connection, which shares its input tributary with other AU3/AU4 conversion cross-connections (i.e. bridged cross-connection), is rolled to a new input tributary, then all cross-connections using the same input tributary is rolled to the new input tributary.

This results in the following updates and notifications:

1. The cross-connection instances sharing the input trib to be rolled are deleted and this is notified via DLT-CRS notifications.
2. New cross-connections using the new input trib are created and this is notified via ENT-CRS notifications.

Identification of VCG tributaries

The system shall allow to roll cross-connections at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

Further Side-effects when rolling low-order cross-connections

If a leg of a low-order cross-connection is being rolled further side-effects as described with ENT-CRS and DLT-CRS may occur.

Example input/output

Example 1:

Suppose the following existing non-BLSR 1-way vc3-cross-connection

1-1-#-#-1-1-3 → 1-1-#-#-5-1-17

is rolled to

1-1-#-#-2-1-3 → 1-1-#-#-5-1-17

also being a non-BLSR cross-connection. This would be accomplished via:

```
ENT-ROLL-VC3:UNITE-NE:1-1-#-#-1-1-3,1-1-#-#-5-1-17:123456:::newinaid=1-1-#-#-2-1-3;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 2:

Suppose the following existing non-BLSR 1-way vc3-cross-connection:

1-1-#-#-1-1-3 → 1-1-#-#-5-1-17

is rolled to:

1-1-#-#-2-1-3 → 1-1-#-#-5-1-17

with the new tributary being involved in a BLRS protection group. This would be accomplished via:

```
ENT-ROLL-VC3:UNITE-NE:1-1-#-#-1-1-3,1-1-#-#-5-1-17:123456:::newinaid=1-1-#-#-2-1-
      3,newloca=BLSR-pprot-name;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 3:

Suppose the following vc4 path-protected cross-connection exists:

worker-leg: 1-1-#-#-1-1-1→ 1-1-#-#-5-1-1

protection leg: 1-1-#-#-2-1-1 → 1-1-#-#-5-1-1

The worker leg now is rolled towards the new in_aid: 1-1-#-#-3-1-1:

This would be accomplished via:

```
ENT-ROLL-VC4:UNITE-NE:1-1-#-#-1-1-1,1-1-#-#-5-1-1:123456:::newinaid=1-1-#-#-3-1-1;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 4:

Suppose the following low-order VC3 bidirectional cross-connection exists:

Input: 1-1-#-#-1-1-1-22

Output: 1-1-#-#-5-1-1-43

The worker leg now is rolled towards the new input: 1-1-#-#-3-1-1-1 which causes the bidirectional cross-connection to be split into two unidirectional ones:

Unchanged leg: from 1-1-#-#-5-1-1-43 to 1-1-#-#-1-1-1-22

Rolled leg: from 1-1-#-#-3-1-1-1 to 1-1-#-#-5-1-1-43

This would be accomplished via:

```
ENT-ROLL-LOVC3:UNITE-NE:1-1-#-#-1-1-1-22,1-1-#-#-5-1-1-43:\
123456:::newinaid=1-1-#-#-3-1-1-1;
```

```
   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

The rolling of a cross-connection can be conceptually understood as the removal of one cross-connection leg followed by the establishment of a new cross-connection leg, with the restriction that the logical output tributary of both legs is the same. Therefore the vaildation rules for the **ENT-ROLL** command are as specified in the DLT-CRS section and the ENT-CRS.

This means that if the **ENT-ROLL** command would fail to meet a criteria as specified in the DLT-CRS section the same command denial would be the consequence. In an anolg way if the **ENT-ROLL** command would fail to meet a criteria as specified in the ENT-CRS section the same command denial would be the consequence.

Validation rule: No Rolling of low-order cross-connections with LOXC/1

For low-order cross-connections, cross-connection command processing shall verify that the low-order switching pack currently used by the system is not of type LOXC/1.

Error code:

`SNVS – Status, Not in Valid State, Roll of low-order cross-connections not allowed with LOXC/1`

Related TL1 messages

- “DLT-CRS: Delete cross-connection” (p. 3-145)
- “ED-CRS: Edit cross-connection” (p. 3-292)
- “ENT-CRS: Enter cross-connection” (p. 3-1464)
- “RTRV-CRS: Retrieve cross-connection” (p. 3-2306)

References

GR-199-CORE

ENT-STM1E notification

Output format

The **ENT-STM1E** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Syntax: **ENT-STM1**:*aid*::*spec_block*

Output parameter

Command parameters

1. **aid** – Access Identifier
   Access Identifier of STM1 port.
   Type: AID.
   Range: SPortAID of STM1E port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is

not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.

6. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

7. **`dexcthr`** – Excessive Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

8. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.

    Range:

    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

13. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.

For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
This identifies the maximum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
maxNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC3 or VC4 (default).
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
This identifies the minimum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
minNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC3 (default) or VC4.
18. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This identifies the unique node identifier of the neighbor node of ports of portClass = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN.
If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is introduced specifically for use by ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This identifies the IP address of the neighbor node of ports of portClass = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is introduced specifically for use by ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic

neighbor detection.
neighborPortID is introduced specifically for use by NN application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE

23. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

24. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

25. **`pmmsfeses`** – Performance Monitoring MS Far-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default: for STM1: 28800.

26. **`pmmsneses`** – Performance Monitoring MS Near-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default: for STM1: 28800.

27. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

28. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: 1 – 64000.
Default: for STM1: 2400.

29. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

30. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

31. **`propagationDelay`** – Propagation Delay
This identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is introduced specifically for use by ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

32. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

33. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
34. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
35. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
36. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
37. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.

38. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
39. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.
40. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).
41. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

42. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.

Range:

- 1: for specific repeating byte (SONET)
- 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

43. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (lso known as the implicit reset method)
- TR-RTR (lso known as the explicit reset method).

Type: Alphanumeric value set.
Range: TR or TRRTR (default).

44. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
    Type: Alphanumeric value set.
    Range: ADAPTIVE, FIXED.
    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

45. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

Example input/output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ENT-STM1E:1-1-#-#-1-
      1::servcond=DISABLE,inputsig=14,...,unequiposig=14"
;
```

ENT-STM1 notification

Output format

The **ENT-STM1** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ENT-STM1**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
Type: AID.
Range: SPortAID of STM1 port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Numeric string.
Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range: 2 to 10.
Default: 7.
3. **brstthr** – Bursty Error Threshold
Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
Type: numeric value set.
Range (as percentage): 5, 10, … 50.
Default: 30.
4. **clientNodeID** – Client Node IDentifier
The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List .
   Type: numeric value set
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **`femspm`** – Far-end Multiplex Section PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.

    Range:

    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

12. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

13. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET.
Type: Alphanumeric value set.
Range: SONET, SDH.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute

represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.

Range:

- YES
- NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 or VC4 (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default) or VC4.

20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This

parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000

    Default: The default values differ for SDH and SONET:

    - STM1: 28800
    - OC3: 154.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

31. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - STM 1: 2400
    - OC3: 155
33. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
34. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
35. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.
36. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **remLinkID** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

38. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

39. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: ″″

40. **rsmstca** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

41. **rspm** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

42. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

43. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

44. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

45. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
   - BURST (default)
   - POISSON.
46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.
48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
 This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
 If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
 Type: Hexadecimal digits.
 Range:
 - 1 byte string
 - 16 bytes string
 - 64 bytes string dependent on the strcwfmt parameter.

 Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
 Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
 The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
 Type: Numeric string.
 Range:
 - 1: for specific repeating byte (SONET)
 - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
 - 16: for 16 byte specific string (ITU/ETSI mode 1).
 - 64: for 64 byte string (SONET).
51. **`tcamd`** – TCA mode
 This parameter shows the modus of the PM thresholding.
 Two thresholding methods are applicable:
 - TR-only (lso known as the implicit reset method)
 - TR-RTR (lso known as the explicit reset method).

 Type: Alphanumeric value set.
 Range: TR or TRRTR (default).

52. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port.
    Type: Alphanumeric value set.
    Range: FIXED.
53. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-STM1:1-1-#-#-1-
      1::servcond=DISABLE,inputsig=14,...,unequiposig=14″
;
```

ENT-STM4 notification

Output format

The **ENT-STM4** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ENT-STM4**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID.
   Range: SPortAID of STM-4 port.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps.
   This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C

6. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

7. **`dexcthr`** – Excessive Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –3; –4; –5.
Default: –3.

8. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

10. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).

12. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C.
Default: 44.

13. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
Type: Alphanumeric value set.
Range: SONET, SDH.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute

represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.

Range:

- YES
- NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4 or VC44C (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4 or VC44C.

20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This

parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **neighborNodeID** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **neighborPortID** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **neighborSigPCID** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **neighborSigPCIPAddr** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **nemspm** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **networkTNAType** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **`networkTNAAddress`** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
  0000000000000000000000000000000000000000…
  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET:

    - STM4: 115200
    - OC12: 615.

30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

31. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
    - STM4: 2400
    - OC12: 616.
33. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
34. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
35. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.
36. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

38. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

39. **`remTID`** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: *""*

40. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
41. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
43. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.

45. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.
48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.

    Range:

    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
50. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.

    Range:

    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
51. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable:

    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

52. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
    Type: Alphanumeric value set.
    Range: ADAPTIVE, FIXED.
    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
53. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C.
    Default: 44.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ENT-STM4:1-1-#-#-1-
      1::servcond=DISABLE,inputsig=44,...,unequiposig=44"
;
```

ENT-STM16 notification

Output format

The **ENT-STM16** notification uses the general format of a **REPT DBCHG** (Report Database Change) notification.

Syntax: **ENT-STM16**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **all** is not allowed.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.

4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute "uniClientNodeID", as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.

Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
10. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU
11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port’s bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
Default: The initial value of X is 4.
Example:
An STM-16 signal might be represented by 93-14-14C-243 which implies 9 VC3s, 1 VC4, 1 VC4-4C, 24 VC3s. The representation is not unique, for example it could also be 33-63-14-14C-243. Worst case of string length for STM16: 47 characters.

13. **istd** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
14. **linkResCls** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
15. **localLinkID** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **localLPID** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

17. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C or VC416C (default)
19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C or VC416C.
20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.

This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **neighborASTNNodeID** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **neighborNodeID** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **neighborPortID** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this

attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number:
0000000000000000000000000000000000000000…
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.
30. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

31. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.
    Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
32. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000
    Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
33. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

34. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

35. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

36. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

37. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

38. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
39. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
40. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
41. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.

Range:

- ENABLE (default): enable consequent action for TIM
- DISABLE: disable consequent action for TIM.

43. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
45. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
46. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

48. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).

49. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
   - 1 byte string
   - 16 bytes string
   - 64 bytes string dependent on the strcwfmt parameter.

   Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 01Hex

50. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
51. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.
    Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).
    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).
52. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port.
    Type: Alphanumeric value set.
    Range: FIXED.
53. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
    Default: 164 = all VC4.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-STM16:1-1-#-#-1-
      1::servcond=DISABLE,inputsig=164,...,unequiposig=164″
;
```

ENT-STM16T notification

Output format

The **ENT-STM16T** notification uses the general format of a REPT DBCHG Report Database Change notification.

Syntax

ENT-STM16T*:aid::spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **all** is not allowed.

Spec block parameters

1. **exprstrc** – Value of Expected J0
   For ENT-STM16T Notification this parameter refers to the incoming (inress) direction.
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **exprstrcout** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex

3. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

4. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SDH.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

6. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

7. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
8. **pmrsneses**out – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
9. **rsmstca** – RS/MS TCA Profile Pointer
   Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
   Type: Alphanumeric string.
   Range: String with maximum of 24 characters.
   Default: DEFAULT0.
10. **rspm** – RS PM Enable
   For ENT-STM16T Notification this parameter refers to the incoming (ingress) direction.
   Enable or disable regenerator section (section) performance monitoring.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).

11. **rspm**out – RS PM Enable in Outgoing Direction
Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
12. **rstimmon** – RS TIM Monitor Enable
For ENT-STM16T this parameter refers to the incoming (Ingress) direction.
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
13. **rstimmonout** – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor
14. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
15. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
16. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.

Range:

- 1: for specific repeating byte (SONET)
- 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

For ENT-STM16T this parameter refers to the incoming (Ingress) direction.

17. **`strcfmtout`** – Format of outgoing J0
For ENT-STM16T Notification this parameter refers to the incoming (Ingress) direction.
The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
Type: Numeric string.
Range:
    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).
18. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).

Example output

```
   UNITE-NE 00-11-20 8:01:03
A  611 REPT DBCHG
   ″Time=8-00-30,DATE=00-11-20:DLT-STM16T:1-1-#-#-4-1″
;
```

ENT-STM64 notification

Output format

Syntax: **ENT-STM64**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID
   Range: STM-64 port AID

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **crstat** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC-64C.
6. **degthr** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **dexcthr** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
8. **exprstrc** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **fec** – FEC Enable
   Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
   Type: Alphanumeric value set.
   Range:

DISABLE (only) for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768/STM256 basis low cost intra office 1310 nm optics.

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

ENABLE (only)for:

- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G

Default:

DISABLE for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768/STM256 basis low cost intra office 1310 nm optics.

ENABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

10. **`femspm`** – Far-end Multiplex Section PM Enable
    Enable or disable of the performance monitoring of the multiplex section.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).

11. **`frcdus`** – Force to DUS
Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
Type: Alphanumeric value set.
Range:
    - ENABLE: (set a forced DUS/DNU)
    - DISABLE (default): (clear a forced DUS/DNU).
12. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
13. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
Default: 644.
14. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
Type: Alphanumeric value set.
Range: SDH.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
15. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute

represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

16. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

17. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C, VC416C or VC464C (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C, VC416C or VC464C.

20. **`msrei`** – MS-REI mode
Definition: An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
Type: Alphanumeric value set.
Range:
   - SINGLE
   - EXTENDED (default).

21. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

22. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.

If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0. If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

23. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

24. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

25. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

27. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

28. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
   - IPV4 (default)
   - IPV6
   - NSAP

29. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:
- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:
- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:
- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

30. **och** – optical channel enable
Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don't have any impact on transmission.
Type: Alphanumeric value set.

Range:

- for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
- for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
- for ports on OP10 and OP40 packs,, which only support OCH: ENABLE: enable optical channel
- for client ports on packs which support OTN: DISABL: disable, optical channel not applicable.

Default:

- for OP10 and OP40 packs, which cannot support OCH: DISABLE
- for OP10 packs, which cannot support OCH: ENABLE
- for OP10 and OP40 packs, which only support OCH: ENABLE
- for client ports on packs which support OTN: DISABLE.

31. **`otnclient`** – OTN Client Port
This parameter indicates whether the port is an OTN Client Port or an optical SDH/SONET port.
Type: Alphanumeric value set.
Range:
    - YES for OTN client port
    - NO for optical SDH/SONET port.

`pmmsfeses` – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459

- SONET OC192: 9835
- SONET OC768: 39340.

32. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-16/OC-48: 1 – 3072000
   - STM-64/OC-192: 1 – 12288000
   - STM-256/OC-768: 1 – 49152000.

   Default: The default values differ for SDH and SONET and the different rates:
   - STM16: 460800
   - STM64: 1843200
   - STM256: 7372800
   - SONET OC48: 2459
   - SONET OC192: 9835
   - SONET OC768: 39340.

33. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

   Default:
   - AUTO for optical SDH/SONET ports
   - NMON for client ports.

34. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
35. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.
36. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
37. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
38. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …255. The unit of this value is ″milliseconds″.

39. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
40. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
41. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
42. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

43. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
44. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
45. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
46. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

47. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in “RTRV-OC192: Retrieve OC192 port” (p. 3-2664)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
48. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
49. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
50. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

51. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
 - 1: for specific repeating byte (SONET)
 - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
 - 16: for 16 byte specific string (ITU/ETSI mode 1).
 - 64: for 64 byte string (SONET).
52. **strcout** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
 - 1 byte string
 - 16 bytes string
 - 64 bytes string dependent on the strcwfmt parameter.

 Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
 Default: 01Hex
53. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
 - 1: for specific repeating byte (SONET)
 - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
 - 16: for 16 byte specific string (ITU/ETSI mode 1).
 - 64: for 64 byte string (SONET).
54. **tcamd** – TCA mode
This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (lso known as the implicit reset method)
- TR-RTR (lso known as the explicit reset method).

Type: Alphanumeric value set.
Range: TR or TRRTR (default).

55. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port.
Type: Alphanumeric value set.
Range: FIXED.

56. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
Default: 644 = all VC4.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-STM64:1-1-#-#-1-1::servcond=DISABLE,och=DISABLE,
inputsig=644,...,unequiposig=644,fec=DISABLE,feccor=DISABLE,fectype=INBAND″
;
```

ENT-STS1 notification

Output format

Syntax: **ENT-STS1**:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – Access Identifier
AID of the SONET tributary. The keyword **all** is not allowed on any level.
Type: AID
Range: STS1NumAID

Spec block parameters

1. **brstintvl**– Bursty Interval
Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
Type: numeric value set.
Range: 2 to 10.
Default: 7.
2. **brstthr** – Bursty Error Threshold
The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
Type: numeric value set.
Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
Default: 30.
3. **crslpbkstat** – Cross-connect loopback status
State of cross- connect loopback.
Type: Alphanumeric value set.
Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
4. **degthr** – Signal Degrade Threshold
The setting of this parameter are used when *sdsfmode* = POISSON
Type: numeric value set.
Range (as exponents of 10): –5; –6 (default); –7; –8; –9
5. **dexcthr** – Excessive Degrade Threshold
The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
Type: numeric value set.
Range (as exponent of 10) –3 (default); –4; –5.

6. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
7. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **`hptimmon`** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`loca`** – Ring A-node TID
   This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
   loca is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in “RTRV-HDR: Retrieve header” (p. 3-2417) description.
   Default: empty string: ″″. This is displayed as \″\″ in output direction.
10. **`locz`** – Ring Z-node TID
   This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.

locz is introduced specifically for use related to ONNS application.
Type: TID or empty-string.
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: "". This is displayed as \"\" in output direction.

11. **neppm** – Near-End Path PM Enable
It indicates whether this performance monitoring feature is enabled or disabled.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
12. **pjepm** – Pointer Justification Event PM Enable
It indicates whether this performance monitoring feature is enabled or disabled.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
13. **pmhpfeses** – Performance Monitoring Path Far-End SES Threshold
This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
14. **pmhpneses** – Performance Monitoring Path Near-End SES Threshold
This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
15. **pse** – PDI-P Switching Enable
The PDI-P Switching Enable parameter indicates if the PDI-P contributes to the automatic path protection switch criteria.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

16. **`ptca`** – Path TCA Profile Pointer
 Type: Alphanumeric string.
 Range: string with up to 24 characters.
 Default: DEFAULT0.
17. **`ptrcrfmt`** – Path Trace Read Format (J1)
 The read format of J1, the incoming path trace (*ptrc*).
 Type: numeric string.
 Range:
 - 2: for non-specific repeating byte
 - 16: for 16 byte specific string.
 - 64: for 64 byte string.
 Default: 2.
18. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
 The degraded state is detected in BURST or POISSON mode.
 Type: Alphanumeric value set.
 Range:
 - BURST
 - POISSON (default).
19. **`substr`** – Substructured
 This parameter indicates if the tributary is substructured.
 Substructure for LO cross-connections:
 A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.
 The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).
 A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **`ED-SUBS`** command.
 Substructure for conversion cross-connections:
 A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.
 The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.
 A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.
 Type: Alphanumeric value set.

Range:

- YES: Automatically sub structured for LO cross-connections
- MANSUB: Manually sub structured for LO cross-connections
- CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
- NO (default): Not sub structured, not used for conversion cross-connections

If the STS-1 tributary is related to a EC1 port, then no LO cross-connections are allowed. As a consequence, the parameter *substr* can only have the values CNV and NO in this case.

20. **`tmonmd`** – Tributary Monitoring Mode
This parameter indicates the mode of the tributary.
Type: Alphanumeric value set.
Range:
    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.
21. **`tribsonetpn`** – Tributary Alarm Severity Assignment Profile
This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

Example output

```
   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ″TIME=14-59-09,DATE=00-10-14:ENT-STS1:1-1-#-#-3-1-1::
      tmonmd=MON,tribsonetpn=Default,...,pmhpneses=30,pmhpfeses=30″
;
```

ENT-T3 notification

Output format

Syntax

`ENT-T3`:*aid::spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
Type: AID
Range: APortAID of DS3 port.

Spec block parameters

1. **`ds3pn`** – DS3 ASAP
DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
2. **`eif`** – Electrical Interface Type
The parameter eif indicates the type of input the port has been provisioned to receive.
Type: Alphanumeric value set.
Range: DS3
3. **`fmt`** – Signal Format
This parameter shows the format for the DS3 signal
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.
Range:
   - C-BIT (default): C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format
   - UNFRAMED-CC: UNFRAMED and CLEAR-CHANNEL format
4. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.

Range:

- YES
- NO (default).

5. **infepm** – Incoming DS3 Far End Path PM Enable
This parameter enables/disables the incoming DS3 far end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
6. **inlnpm** – Incoming DS3 Line PM Enable
This parameter enables/disables the incoming DS3 line performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
7. **inpm** – Incoming DS3 Near End Path PM Enable
This parameter enables/disables the incoming DS3 near end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
8. **lbo** – Line Build Out
This parameter defines the cable length applicable for the DS3 port.
Type: Alphanumeric value set.
Range:
    - SHORT (default): 0 … 68.5 m (0 … 225 ft)
    - LONG: 68.5 … 137 m (225 … 450 ft).
9. **lpbk** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
10. **outmntc** – Outgoing Maintenance Signal
This parameter determines whether an AIS, a DS3AIS or IDLE signal is transmitted in response to an unconnected input.
Type: Alphanumeric value set.
Range: AIS (default), DS3AIS, IDLE

11. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
    This parameter enables/disables the outgoing DS3 far end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
12. **`outpm`** – Outgoing DS3 Near End Path PM Enable
    This parameter enables/disables the outgoing DS3 near end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
13. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
14. **`pmsesl`** – PM Threshold for SES – Line
    PM Threshold (errors in a second) for declaring SES – Line
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
15. **`pmsesp`** – PM Threshold for SES – Path Incoming
    PM Threshold (errors in a second) for declaring SES – Path in incoming direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
16. **`pmsespfe`** – PM Threshold for Far-End SES – Path Incoming
    PM Threshold (errors in a second) for declaring Far-End SES – Path in incoming direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
17. **`pmsespfeout`** – PM Threshold (errors in a second) for declaring Far-End SES – Path in outgoing direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44

18. **pmsespout** – PM Threshold for SES – Path Outgoing
PM Threshold (errors in a second) for declaring SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44
19. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-3 to 10exp-9.
Type: numeric value set.
Range: (as exponents of 10): -3, -4, -5, -6, -7, -8, -9
Default: -6
20. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVIC: Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

    Default: INSERVICE.
21. **tca** – DS3 PM TCA
The parameter *tca* assigns a DS3 PM TCA profile name to the designated AID.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters: DEFAULT, DEFAULT0
Default: DEFAULT0
22. **vmr** – Violation Monitoring Mode
This parameter indicates the status of the DS3 P-bit parity violation monitoring and removal.
Type: Alphanumeric value set. A hyphen ("-") is allowed in the alphanumeric values additionally.
Range: VMR, VM, NO-VM
Default: NO-VM.

Example output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-T3:1-1-#-#-21-1::
      cbitmis=NORMAL,eif=DS3,ptype=T3,fmt=C-BIT,.... sigmis=NORMAL,vmr=NO-VM ″
;
```

ENT-TCA-PROF: Enter TCA Profile

The **ENT-TCA-PROF** command allows the user to create a new Performance Monitoring (PM) Threshold Crossing Alert (TCA) profile with a specific name. The system supports two default sets of profiles called ″Default″ and ″Default0″. When a new profile is created, all the threshold values are copied from the ″Default″ profile.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM3

Priority: 1

Abortable: Yes

Input format

Command syntax:

ENT-TCA-PROF:*tid::ctag::pftype,pfname;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
4. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Output format

If the NE fully complies with the ENT-TCA-PROF request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command shall establish/create a new TCA profile of type as specified by means of parameter pftype. The name of the new TCA profile corresponds to the parameter pfname. The creation of the new TCA profile is reported by means of an ENT-TCA-PROF notification.

Example input/output

The following example shows the successful completion of a ENT-TCA-PROF command by the NE:

```
ENT-TCA-PROF:UNITE-NE::123456::section-line,DTAG;
UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command Error responses section.

Validation rule: Maximum number of TCA profiles reached

If the maximum number of TCA profiles reached, the error message is generated.

Error code:

`SSRE – Status, System Resources Exceeded, limit of profiles exceeded`

Validation rule: profile name already exists

If the system receives a new profile name that already exists for that profile type, the following error response is returned.

Error code: `IEAE – Input, Entity Already Exists, duplicate profile name`

Related TL1 messages

- "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982)
- "DLT-TCA-PROF: Delete TCA profile" (p. 3-259)
- "ED-TCA-PROF: Edit TCA profile" (p. 3-1214)

References

GR-833-CORE

ENT-TCA-PROF notification

Output format

Command syntax: **ENT-TCA-PROF**::*pftype*,*pfname*:*spec_block;*

Output parameters

Command parameters

1. **pftype** – Profile Type
This indicates the type of PM TCA profile for the selected facility.
Type: value set.
Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
2. **pfname** – Profile Name
This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
Type: Alphanumeric string.
Range: This is an alphanumeric character string with a maximum of 24 characters.
Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Spec block parameters - profile parameters of type SECTION-LINE

1. **qhcvsoc3**
15-minute threshold for CV-S bin for OC-3.
Type: Integer.
Range: 0-57599100
Default: 140.
2. **daycvsoc3**
Day threshold for CV-S bin of OC-3.
Type: Integer.
Range: 0-4294967295
Default: 1344.
3. **qhcvsoc12**
15-minute threshold for CV-S bin for OC-12.
Type: Integer.
Range: 0-57599100
Default: 560.

4. **daycvsoc12**
   Day threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-4294967295
   Default: 5376.
5. **qhcvsoc48**
   15-minute threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-57599100
   Default: 2240.
6. **daycvsoc48**
   Day threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-4294967295
   Default: 21504.
7. **qhcvsoc192**
   15-minute threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-57599100
   Default: 8960.
8. **daycvsoc192**
   Day threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-4294967295
   Default: 86016.
9. **qhcvsoc768**
   15-minute threshold for CV-S bin for OC-768.
   Type: Integer.
   Range: 0-57599100
   Default: 35840.
10. **daycvsoc768**
    Day threshold for CV-S bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 344064.
11. **qhcvsec1**
    15-minute threshold for CV-S bin for EC-1.
    Type: Integer.
    Range: 0-57599100
    Default: 47

12. **daycvsec1**
    Day threshold for CV-S bin of EC-1.
    Type: Integer.
    Range: 0-4294967295
    Default: 448
13. **qhess**
    15-minute threshold for ES-S bin.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **dayess**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **qhsess**
    15-minute threshold for SES-S bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
16. **daysess**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.
17. **qhsefss**
    15-minute threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
18. **daysefss**
    Day threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.
19. **qhcvloc3**
    15-minute threshold for CV-L and CV-LFE bin for OC-3.
    Type: Integer.
    Range: 0-172799100
    Default: 140.

20. **`daycvloc3`**
    Day threshold for CV-L bin and CV-LFE for OC-3.
    Type: Integer.
    Range: 0-4294967295
    Default: 1344.
21. **`qhcvloc12`**
    15-minute threshold for CV-L and CV-LFE bin for OC-12.
    Type: Integer.
    Range: 0-691199100
    Default: 560.
22. **`daycvloc12`**
    Day threshold for CV-L bin and CV-LFE for OC-12.
    Type: Integer.
    Range: 0-4294967295
    Default: 5376.
23. **`qhcvloc48`**
    15-minute threshold for CV-L and CV-LFE bin for OC-48.
    Type: Integer.
    Range: 0-2764799100
    Default: 2240.
24. **`daycvloc48`**
    Day threshold for CV-L and CV-LFE bin for OC-48.
    Type: Integer.
    Range: 0-4294967295
    Default: 21504.
25. **`qhcvloc192`**
    15-minute threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 8960.
26. **`daycvloc192`**
    Day threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 86016.
27. **`qhcvloc768`**
    15-minute threshold for CV-L and CV-LFE bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 35840.

28. **`daycvloc768`**
Day threshold for CV-L and CV-LFE bin for OC-768.
Type: Integer.
Range: 0-4294967295
Default: 344064.

29. **`qhcvlec1`**
15-minute threshold for CV-L and CV-LFE bin for EC-1.
Type: Integer.
Range: 0-57599100
Default: 47

30. **`daycvlec1`**
Day threshold for CV-L bin and CV-LFE for EC-1.
Type: Integer.
Range: 0-4294967295
Default: 448

31. **`qhesl`**
15-minute threshold for ES-L and ES-LFE bin.
Type: Integer.
Range: 0-900
Default: 25.

32. **`dayesl`**
Day threshold for ES-L and ES-LFE bin.
Type: Integer.
Range: 0-86400
Default: 250.

33. **`qhfeccloc192`**
15-minute threshold for FECC-L bin for OC-192.
Type: Integer.
Range: 0 – 4294967295
Default: 5184.

34. **`dayfeccloc192`**
Day threshold for FECC-L bin for OC-192.
Type: Integer.
Range: 0 – 4294967295
Default: 497644.

35. **`qhfeccloc768`**
15-minute threshold for FECC-L bin for OC-768.
Type: Integer.
Range: 0 – 4294967295
Default: 20736.

36. **`dayfeccloc768`**
    Day threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
37. **`qhsesl`**
    15-minute threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
38. **`daysesl`**
    Day threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.
39. **`qhuasl`**
    15-minute threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
40. **`dayuasl`**
    Day threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

Spec block parameters - profile parameters of type RS-MS

1. **`qhrsnbbe`**
   15-minute threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 9000.
2. **`dayrsnbbe`**
   Day threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
3. **`qhrsnes`**
   15-minute threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 180.

4. **dayrsnes**
   Day threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.
5. **qhrsnses**
   15-minute threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
6. **dayrsnses**
   Day threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 20.
7. **qhrsnuas**
   15-minute threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayrsnuas**
   Day threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 20.
9. **qhmsbbestm1**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 228000.
10. **daymsbbestm1**
    Day threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 432000.
11. **qhmsbbestm4**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 0-691199100
    Default: 912000.

12. **daymsbbestm4**
Day threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
Type: Integer.
Range: 00-4294967295
Default: 1728000.

13. **qhmsbbestm16**
15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
Type: Integer.
Range: 0-2764799100
Default: 3648000.

14. **daymsbbestm16**
Day threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
Type: Integer.
Range: 0-4294967295
Default: 6912000.

15. **qhmsbbestm64**
15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
Type: Integer.
Range: 0-4294967295
Default: 14592000.

16. **daymsbbestm64**
Day threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
Type: Integer.
Range: 0-4294967295
Default: 27648000.

17. **qhmsbbestm256**
15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
Type: Integer.
Range: 0-4294967295
Default: 58368000.

18. **daymsbbestm256**
Day threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
Type: Integer.
Range: 0-4294967295
Default: 110592000.

19. **qhmses**
15-minute threshold for MS_N_ES and MS_F_ES bin.
Type: Integer.
Range: 0-900
Default: 50.

20. **`daymses`**
    Day threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.
21. **`qhmsnfeccstm64`**
    15-minute threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 5184.
22. **`daymsnfeccstm64`**
    Day threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 49764.
23. **`qhmsnfeccstm256`**
    15-minute threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
24. **`daymsnfeccstm256`**
    Day threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
25. **`qhmsses`**
    15-minute threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
26. **`daymsses`**
    Day threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-77760
    Default: 15.

27. **qhmsuas**
    15-min threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
28. **daymsuas**
    Day threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters - reset profile parameters of type RS-MS

1. **qhrtrrsnbbe**
   15-minute reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **dayrtrrsnbbe**
   Day reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
3. **qhrtrrsnes**
   15-minute reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 20.
4. **dayrtrrsnes**
   Day reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **qhrtrrsnses**
   15-minute reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrrsnses**
   Day reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 0.

7. **`qhrtrrsnuas`**
   15-minute reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **`dayrtrrsnuas`**
   Day reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 0.
9. **`qhrtrmsbbestm1`**
   15-min reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 2400.
10. **`dayrtrmsbbestm1`**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 43200.
11. **`qhrtrmsbbestm4`**
   15-min reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 0-691199100
   Default: 9600.
12. **`dayrtrmsbbestm4`**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 172800.
13. **`qhrtrmsbbestm16`**
   15-min reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
   Type: Integer.
   Range: 0-2764799100
   Default: 38400.
14. **`dayrtrmsbbestm16`**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 691200.

15. **qhrtrmsbbestm64**
    15-min reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 153600.
16. **dayrtrmsbbestm64**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 2764800.
17. **qhrtrmsbbestm256**
    15-min reset threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 614400.
18. **dayrtrmsbbestm256**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 11059200.
19. **qhrtrmses**
    15-minute reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
20. **dayrtrmses**
    Day reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 15.
21. **qhrtrmsnfeccstm64**
    15-minute reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
22. **dayrtrmsnfeccstm64**
    Day reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.

23. **qhrtrmsnfeccstm256**
15-minute reset threshold for MS-N-FECC bin forSTM-256.
Type: Integer.
Range: 0 – 4294967295
Default: 0.

24. **dayrtrmsnfeccstm256**
Day reset threshold for MS-N-FECC bin forSTM-256.
Type: Integer.
Range: 0 – 4294967295
Default: 0.

25. **qhrtrmsses**
15-minute reset threshold for MS_N_SES and MS_F_SES bin.
Type: Integer.
Range: 0-810
Default: 0.

26. **dayrtrmsses**
Day reset threshold for MS_N_SES and MS_F_SES bin.
Type: Integer.
Range: 0-77760
Default: 0.

27. **qhrtrmsuas**
15-min reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-900
Default: 0.

28. **dayrtrmsuas**
Day reset threshold for MS_N_UAS and MS_F_UAS.
Type: Integer.
Range: 0-86400
Default: 0.

Spec block parameters - profile parameters of type ODU

1. **qhodupbbeodu1**
15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-18377100
Default: 32000.

2. **dayodupbbeodu1**
Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
Type: Integer.
Range: 0-1764201600
Default: 61000.

3. **qhodupes**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 50.
4. **dayodupes**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P..
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **qhodupses**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 10.
6. **dayodupses**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 15.
7. **qhodupuas**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayodupuas**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 20.

Spec block parameters - reset profile parameters of type ODU

1. **qhrtrodupbbeodu1**
   15-minute reset threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-18377100
   Default: 200.
2. **dayrtrodupbbeodu1**
   Day reset threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-1764201600
   Default: 4800.

3. **qhrtrodupes**
   15-minute reset threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 5.
4. **dayrtrodupes**
   Day reset threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 15.
5. **qhrtrodupses**
   15-minute reset threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrodupses**
   Day reset threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 0.
7. **qhrtrodupuas**
   15-minute threset reshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **dayrtrodupuas**
   Day reset threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 0.

Spec block parameters - profile parameters of type HO-VC

1. **qhvcbbevc3**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
2. **dayvcbbevc3**
   Day threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

3. **qhvcbbevc4**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbevc4**
   Day threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcbbevc44c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
6. **dayvcbbevc44c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
7. **qhvcbbevc416c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
8. **dayvcbbevc416c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
9. **qhvcbbevc464c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
10. **dayvcbbevc464c**
    Day threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
    Type: Integer.
    Range: 0-691113600
    Default: 48000.

11. **qhvcesvc3**
    15-minute threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 150.
12. **dayvcesvc3**
    Day threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 600.
13. **qhvcesvc4**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
14. **dayvcesvc4**
    Day threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.
15. **qhvcesvc44c**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
16. **dayvcesvc44c**
    Day threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.
17. **qhvcesvc416c**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
18. **dayvcesvc416c**
    Day threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.

19. **qhvcesvc464c**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
20. **dayvcesvc464c**
    Day threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.
21. **qhvcsesvc3**
    15-minute threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
22. **dayvcsesvc3**
    Day threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
23. **qhvcsesvc4**
    15-minute threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
24. **dayvcsesvc4**
    Day threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
25. **qhvcsesvc44c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
26. **dayvcsesvc44c**
    Day threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Range: 0-77760
    Default: 20.

27. **qhvcsesvc416c**
15-minute threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-810
Default: 15.
28. **dayvcsesvc416c**
Day threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-77760
Default: 20.
29. **qhvcsesvc464c**
15-minute threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-810
Default: 15.
30. **dayvcsesvc464c**
Day threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-77760
Default: 20.
31. **qhvcuasvc3**
15-minute threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-900
Default: 15.
32. **dayvcuasvc3**
Day threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-86400
Default: 20.
33. **qhvcuasvc4**
15-minute threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-900
Default: 15.
34. **dayvcuasvc4**
Day threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 20.

35. **qhvcuasvc44c**
 15-minute threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
 Type: Integer.
 Range: 0-900
 Default: 15.
36. **dayvcuasvc44c**
 Day threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
 Type: Integer.
 Range: 0-86400
 Default: 20.
37. **qhvcuasvc416c**
 15-minute threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
 Type: Integer.
 Range: 0-900
 Default: 15.
38. **dayvcuasvc416c**
 Day threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
 Type: Integer.
 Range: 0-86400
 Default: 20.
39. **qhvcuasvc464c**
 15-minute threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
 Type: Integer.
 Range: 0-900
 Default: 15.
40. **dayvcuasvc464c**
 Day threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
 Type: Integer.
 Range: 0-86400
 Default: 20.
41. **qhpjcpos**
 15-minute threshold for positive PJC bin.
 Type: Integer.
 Range: 0-1800000
 Default: 30.
42. **daypjcpos**
 Day threshold for positive PJC bin.
 Type: Integer.
 Range: 0-172800000
 Default: 2880.

43. **qhpjcneg**
 15-minute threshold for negative PJC bin.
 Type: Integer.
 Range: 0-1800000
 Default: 30.
44. **daypjcneg**
 Day threshold for negative PJC bin.
 Type: Integer.
 Range: 0-172800000
 Default: 2880.

Spec block parameters - reset profile parameters of type HO-VC

1. **qhrtrvcbbevc3**
 15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
 Type: Integer.
 Range: 0-7199100
 Default: 200.
2. **dayrtrvcbbevc3**
 Day reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
 Type: Integer.
 Range: 0-691113600
 Default: 4800.
3. **qhrtrvcbbevc4**
 15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
 Type: Integer.
 Range: 0-7199100
 Default: 200.
4. **dayrtrvcbbevc4**
 Day reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
 Type: Integer.
 Range: 0-691113600
 Default: 4800.
5. **qhrtrvcbbevc44c**
 15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
 Type: Integer.
 Range: 0-7199100
 Default: 200.
6. **dayrtrvcbbevc44c**
 Day reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
 Type: Integer.
 Range: 0-691113600
 Default: 4800.

7. **`qhrtrvcbbevc416c`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
8. **`dayrtrvcbbevc416c`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
9. **`qhrtrvcbbevc464c`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
10. **`dayrtrvcbbevc464c`**
    dayrtrvcbbevc464c
    Type: Integer.
    Range: 0-691113600
    Default: 4800.
11. **`qhrtrvcesvc3`**
    15-minute reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
12. **`dayrtrvcesvc3`**
    Day reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 60.
13. **`qhrtrvcesvc4`**
    15-minute reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 20.
14. **`dayrtrvcesvc4`**
    Day reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.

15. **qhrtrvcesvc44c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-900
Default: 20.
16. **dayrtrvcesvc44c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-86400
Default: 150.
17. **qhrtrvcesvc416c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-900
Default: 20.
18. **dayrtrvcesvc416c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-86400
Default: 150.
19. **qhrtrvcesvc464c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 20.
20. **dayrtrvcesvc464c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 150.
21. **qhrtrvcsesvc3**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-810
Default: 0.
22. **dayrtrvcsesvc3**
Day reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
Type: Integer.
Range: 0-77760
Default: 0.

23. **qhrtrvcsesvc4**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-810
Default: 0.

24. **dayrtrvcsesvc4**
Day reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
Type: Integer.
Range: 0-77760
Default: 0.

25. **qhrtrvcsesvc44c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
Type: Integer.
Range: 0-810
Default: 0.

26. **dayrtrvcsesvc44c**
Day reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
Type: Integer.
Range: 0-77760
Default: 0.

27. **qhrtrvcsesvc416c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-810
Default: 0.

28. **dayrtrvcsesvc416c**
Day reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
Type: Integer.
Range: 0-77760
Default: 0.

29. **qhrtrvcsesvc464c**
15-minute reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-810
Default: 0.

30. **dayrtrvcsesvc464c**
Day reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
Type: Integer.
Range: 0-77760
Default: 0.

31. **qhrtrvcuasvc3**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-900
Default: 0.

32. **dayrtrvcuasvc3**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
Type: Integer.
Range: 0-86400
Default: 0.

33. **qhrtrvcuasvc4**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-900
Default: 0.

34. **dayrtrvcuasvc4**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 0.

35. **qhrtrvcuasvc44c**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
Type: Integer.
Range: 0-900
Default: 0.

36. **dayrtrvcuasvc44c**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
Type: Integer.
Range: 0-86400
Default: 0.

37. **qhrtrvcuasvc416c**
15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
Type: Integer.
Range: 0-900
Default: 0.

38. **dayrtrvcuasvc416c**
Day reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
Type: Integer.
Range: 0-86400
Default: 0.

39. **`qhrtrvcuasvc464c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
40. **`dayrtrvcuasvc464c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
41. **`qhrtrpjcpos`**
    15-minute reset threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
42. **`dayrtrpjcpos`**
    Day reset threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.
43. **`qhrtrpjcneg`**
    15-minute reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
44. **`dayrtrpjcneg`**
    Day reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.

Spec block parameters - profile parameters of type PATH

1. **`qhcvpsts1`**
   15-minute threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-57599100
   Default: 15.
2. **`daycvpsts1`**
   Day threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-4294967295
   Default: 125.

3. **`qhcvpsts3c`**
   15-minute threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 0-57599100
   Default: 25.
4. **`daycvpsts3c`**
   Day threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 4294967295
   Default: 250.
5. **`qhcvpsts12c`**
   15-minute threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 0-57599100
   Default: 75.
6. **`daycvpsts12c`**
   Day threshold for threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 4294967295
   Default: 750.
7. **`qhcvpsts48c`**
   15-minute threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 0-57599100
   Default: 300.
8. **`daycvpsts48c`**
   Day threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 4294967295
   Default: 3000.
9. **`qhcvpsts192c`**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-57599100
   Default: 1200.
10. **`daycvpsts192c`**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-4294967295
   Default: 12000.

11. **qhespsts1**
    15-minute threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-900
    Default: 12.
12. **dayespsts1**
    Day threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-86400
    Default: 100.
13. **qhespsts3c**
    15-minute threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-900
    Default: 20.
14. **dayespsts3c**
    Day threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-86400
    Default: 200.
15. **qhespsts12c**
    15-minute threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-900
    Default: 60.
16. **dayespsts12c**
    Day threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-86400
    Default: 600.
17. **qhespsts48c**
    15-minute threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-900
    Default: 240.
18. **dayespsts48c**
    Day threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-86400
    Default: 2400.

19. **`qhespsts192c`**
Day threshold for ES-P and ES-PFE bin for STS192C.
Type: Integer.
Range: 0-900
Default: 240.

20. **`dayespsts192c`**
Day threshold for ES-P and ES-PFE bin for STS192C.
Type: Integer.
Range: 0-86400
Default: 9600.

21. **`qhsesp`**
15-minute threshold for SES-P and SES-PFE bin.
Type: Integer.
Range: 0-810
Default: 3.

22. **`daysesp`**
Day threshold for SES-P and SES-PFE bin.
Type: Integer.
Range: 0-77760
Default: 7.

23. **`qhuasp`**
15-minute threshold for UAS-P and UAS-PFE bin.
Type: Integer.
Range: 0-900
Default: 10.

24. **`dayuasp`**
Day threshold for UAS-P and UAS-PFE bin.
Type: Integer.
Range: 0-86400
Default: 10.

25. **`qhppjcgen`**
15-minute threshold for PPJC-PGen bin.
Type: Integer.
Range: 0-1800000
Default: 30.

26. **`dayppjcgen`**
Day threshold for PPJC-PGen bin.
Type: Integer.
Range: 0-172800000
Default: 2880.

27. **qhnpjcgen**
    15-minute threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
28. **daynpjcgen**
    Day threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
29. **qhppjcdet**
    15-minute threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
30. **dayppjcdet**
    Day threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
31. **qhnpjcdet**
    15-minute threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-18000000
    Default: 30.
32. **daynpjcdet**
    Day threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Spec block parameters - profile parameters of type ENET

1. **qhpdege**
   15-minute threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 187500.
2. **daypdege**
   Day threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.

3. **`qhpdevcg`**
   15-minute threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 10.
4. **`daypdevcg`**
   Day threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
5. **`qhedfe1ge`**
   15min EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 1875000.
6. **`dayedfe1ge`**
   24 h EDFE for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.
7. **`qhedfc1ge`**
   15min EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 10.
8. **`dayedfc1ge`**
   24 h EDFC for 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
9. **`qhedfe10ge`**
   15min EDFE for 10GbE Ethernet port.
   Type: Integer.
   Range: 0-2147483647
   Default: 18750000.
10. **`dayedfe10ge`**
    24 h EDFE for 10GbE Ethernet port
    Type: Integer.
    Range: 0-4294967295
    Default: 90000000.

11. **qhedfc10ge**
    15min EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
12. **dayedfc10ge**
    24 h EDFC for 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.
13. **qhedfe1gevcg**
    15min EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
14. **dayedfe1gevcg**
    24 h EDFE for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
15. **qhedfc1gevcg**
    15min EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
16. **dayedfc1gevcg**
    24 h EDFC for 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
17. **qhedfe10gevcg**
    15min EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
18. **dayedfe10gevcg**
    24 h EDFE for 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 500.

19. **`qhedfc10gevcg`**
    15min EDFC for 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
20. **`dayedfc10gevcg`**
    24 h EDFC for 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295-4294967295
    Default: 500.

Spec block parameters - reset profile parameters of type ENET

1. **`qhrtrpdege`**
   15-minute reset threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 937500.
2. **`dayrtrpdege`**
   Day reset threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 4500000.
3. **`qhrtrpdevcg`**
   15-minute reset threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 2.
4. **`dayrtrpdevcg`**
   Day reset threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
5. **`qhrtredfe1ge`**
   15min reset threshold for EDFE of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 937500.
6. **`dayrtredfe1ge`**
   24h reset threshold for EDFE of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 4500000.

7. **`qhrtredfc1ge`**
   15min reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 2.
8. **`dayrtredfc1ge`**
   24h reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
9. **`qhrtredfe10ge`**
   15min reset threshold for EDFE of 10GbE Ethernet port.
   Type: Integer.
   0-2147483647
   Default: 9375000.
10. **`dayrtredfe10ge`**
    24h reset threshold for EDFE of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 45000000.
11. **`qhrtredfc10ge`**
    15min reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    0-2147483647
    Default: 20.
12. **`dayrtredfc10ge`**
    24h reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
13. **`qhrtredfe1gevcg`**
    15min reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
14. **`dayrtredfe1gevcg`**
    24h reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.

15. **qhrtredfc1gevcg**
    15min reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
16. **dayrtredfc1gevcg**
    24h reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
17. **qhrtredfe10gevcg**
    15min reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    0-2147483647
    Default: 20.
18. **dayrtredfe10gevcg**
    24h reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    Range: 0-294967295
    Default: 50.
19. **qhrtredfc10gevcg**
    15min reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
20. **dayrtredfc10gevcg**
    24h reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

Spec block parameters- profile parameters of type LOVC

1. **qhvcbbevc12**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 9000.
2. **dayvcbbevc12**
   Day threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 12000.

3. **qhvcbbelovc3**
   15-minute threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbelovc3**
   Day threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcesvc12**
   15-minute threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 120.
6. **dayvcesvc12**
   Day threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 350.
7. **qhvceslovc3**
   15-minute threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 150.
8. **dayvceslovc3**
   Day threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 600.
9. **qhvcsesvc12**
   15-minute threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
10. **dayvcsesvc12**
    Day threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.

11. **`qhvcseslovc3`**
    15-minute threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
12. **`dayvcseslovc3`**
    Day threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
13. **`qhvcuasvc12`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
14. **`dayvcuasvc12`**
    Day threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
15. **`qhvcuaslovc3`**
    15-minute threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
16. **`dayvcuaslovc3`**
    Day threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Spec block parameters- reset profile parameters of type LOVC

1. **`qhrtrvcbbevc12`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 50.
2. **`dayrtrvcbbevc12`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 1200.

3. **`qhrtrvcbbelovc3`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **`dayrtrvcbbelovc3`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
5. **`qhrtrvcesvc12`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 5.
6. **`dayrtrvcesvc12`**
   Day reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 35.
7. **`qhrtrvceslovc3`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 10.
8. **`dayrtrvceslovc3`**
   Day reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 60.
9. **`qhrtrvcsesvc12`**
   15-minute reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
10. **`dayrtrvcsesvc12`**
    Day reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.

11. **qhrtrvcseslovc3**
    15-minute reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
12. **dayrtrvcseslovc3**
    Day reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
13. **qhrtrvcuasvc12**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
14. **dayrtrvcuasvc12**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
15. **qhrtrvcuaslovc3**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
16. **dayrtrvcuaslovc3**
    Day reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

Spec block parameters.- Profile Parameters of Type LOPATH

1. **qhcvvt1**
   15-minute threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-14399100
   Default: 75.
2. **daycvvt1**
   Day threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-1382313600
   Default: 750.

3. **qhesvt1**
   15-minute threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
4. **dayesvt1**
   Day threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.
5. **qhsesvt1**
   15-minute threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-810
   Default: 40.
6. **daysesvt1**
   Day threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-77760
   Default: 900.
7. **qhuasvt1**
   15-minute threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
8. **dayuasvt1**
   Day threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.

Spec block parameters - profile parameters of type DS3

1. **qhcvl**
   15-minute threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0-7200000
   Default: 387.
2. **daycvl**
   Day threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0- 691200000
   Default: 3865.

3. **`qhesl`**
   15-minute threshold for Es-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
4. **`dayesl`**
   Day threshold for ES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
5. **`qhesal`**
   15-minute threshold for ESA-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
6. **`dayesal`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
7. **`qhesbl`**
   15-minute threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
8. **`dayesbl`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
9. **`qhsesl`**
   15-minute threshold for SES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-810
   Default: 4.
10. **`daysesl`**
    Day threshold for SES-L bin of DS3 Line.
    Type: Integer.
    Range: 0-77760
    Default: 40.

11. **qhcvp**
    15-minute threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-7200000
    Default: 382.
12. **daycvp**
    Day threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-691200000
    Default: 3820.
13. **qhesp**
    15-minute threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **dayesp**
    Day threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **qhesap**
    15-minute threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
16. **dayesap**
    Day threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
17. **qhesbp**
    15-minute threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.

18. **dayesbp**
    Day threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
19. **qhsesp**
    15-minute threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-810
    Default: 4.
20. **daysesp**
    Day threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-77760
    Default: 40.
21. **qhuasp**
    15-minute threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 10.
22. **dayuasp**
    Day threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 10.
23. **qhsasp**
    15-minute threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
24. **daysasp**
    Day threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 8.

Example output

The following example reports the creation of the TCA profile DTAG:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ”TIME=15-00-00,DATE=00-11-03:ENT-TCA-PROF::RS-MS,DTAG:
qhrsnes=10,qhrsnbbe=6”
;
```

ENT-TMAT3 notification

Output format

Command syntax: **ENT-TMAT3**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **aid** – Access Identifier of a TransMUX asynchronous adjunct port.
Type: AID
Range: TMuxAPortAID
The keyword **all** is allowed up to shelf level.
Default: None.

Spec block parameters

1. **ds3pn** – DS3 ASAP
DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
2. **fmt** – Signal Format
This parameter shows the format for the DS3 signal
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
Range:
   - C-BIT: C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format. M23 format contains seven DS2 signals, which contain DS1 signals.

   Default: C-BIT.
3. **outfepm** – Outgoing DS3 Far End Path PM Enable
This parameter enables/disables the outgoing DS3 far end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE.
Default: DISABLE.
4. **outpm** – Outgoing DS3 Near End Path PM Enable
This parameter enables/disables the outgoing DS3 near end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE.
Default: DISABLE.

5. **`pmode`** – Port Mode
   This parameter sets the mode of the port.
   Type: Alphanumeric value set.
   Range:
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
   Default: NMON.
6. **`servcond`** – Service Condition
   This parameter indicates its status concerning service involvement.
   Type: Alphanumeric value set.
   Range:
   - INSERVICE: Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
   Default: INSERVICE.
7. **`tca`** – DS3 PM TCA
   This parameter assigns a DS3 PM TCA profile name to the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   - DEFAULT
   - DEFAULT0
   Default: DEFAULT0.

Example input/output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   ″TIME=08-00-12,DATE=00-11-28:ENT-TMAT3:1-1-#-#-21-tma1:: eif=DS3,fmt=C-
      BIT,....,vmr=NO-VM ″
;
```

ENT-TMST3 notification

Output format

Command syntax: **`ENT-TMST3`**:*aid*::*spec_block*;

Output parameters

Command parameters

1. **`aid`** – Access Identifier of a TransMUX virtual synchronous adjunct port.
   Type: AID
   Range: TMuxSPortAID
   The keyword **`all`** is allowed up to shelf level.
   Default: None.

Example input/output

```
   UNITE-NE 00-11-28 08:00:30
A  001 REPT DBCHG
   "TIME=08-00-12,DATE=00-11-28:ENT-TMST3:1-1-#-#-21-tms1"
;
```

ENT-ULS: Enter upper layer stack

The **ENT-ULS** command is used to provision parameters of the TCP/IP stack.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-ULS**:*tid::ctag:::spec_block;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of tid in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of ctag in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **ip_ad_astn** – IP Address of ASTN
   This parameter is used to set the router ID for the SCN and the SNN node.
   Note: The provisioning of this IP address is ONLY ALLOWED ONCE.
   Type: Four dot separated numeric strings.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.

   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – IP address for UNI
   This parameter is used to set the UNI node IP address.
   Type: Four dot separated numeric strings.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.
3. **sigPCIP_Address** – IP address for Signaling Controller (E-NNI node)
   This parameter is used to set the signaling controller IP address (E-NNI node address).
   Type: Four dot separated numeric strings.

Range:

- For first numeric string: 0 .. 223
- For second to fourth numeric string: 0 .. 255.

The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.

4. **sigPCID** – Identifier of the Signaling Controller (E-NNI node)
   This parameter is used to set the signaling controller identifier.
   Type: Four dot separated numeric strings.
   Range: Four times numeric string: 0 .. 255
5. **ip_drad** – IP Default Router Address
   This parameter is used to set or retrieve the IP default router address (IPv4) to which the host points to, needed for the T-TD functionality.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
6. **ip_drlan** – IP Default Router LAN port AID
   This parameter specifies with which LAN port the default router is associated.
   Type: AID.
   Range: LanPortAID. The value lanall is not allowed.
   Default: 1-1-lan1.
7. **lv_port** – T-TD Length Value Port
   This parameter is used to set or retrieve the port number over which the T-TD function is accessible using the Length Value encoding method.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3081.
8. **raw_port** – T-TD Raw Mode Port
   This parameter is used to set or retrieve the port AID of number over which the T-TD function is accessible using raw-mode encoding.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3082.

Output format

If the ENT-ULS request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command setsthe parameters as specified for the input spec_block parameters.

Reset: Modification of any of the parameters *ip_ad_astn*, *lv_port*, *raw_port* leads to a reset of the DCF controller and disconnection of management communications.

Implicit UNI node creation

Setting the parameter *ip_ad_uni* away from the default value creates the UNI node.

Implicit E-NNI node creation

Setting the parameter *sigPCIP_Address* away from the default value creates the E-NNI node. There is no implicit or explicit command to delete an E-NNI node.

Example input/output

The following example shows the result of an ENT-ULS command, which modifies the port number used for T-TD length value encoding:

```
ENT-ULS:LT-UNITE::123456:::lv_port=7777;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ENT-ULS** command.

Validation rule: no TCP port number change with enabled T-TD function

T-TD related port numbers cannot be changed as long as T-TD function is enabled.

Error code:

`SNVS – Status, Not in Valid State, port numbers cannot be changed when T-TD is enabled`

Validation rule: check for correct IP configuration

The **ENT-ULS** command is only allowed if there is already a valid IP configuration (ip_ad – IP address of LAN, ip_snm – subnet mask of LAN) provisioned for the LAN port specified by ip_drlan.

Error code: `SNVS – Status, Not in Valid State, incomplete IP configuration`

Validation rule: check for correct configured Default Router LAN Port

The system do not allow to change the default router address (ip_drad) as long as there is no default router LAN port (ip_drlan) specified which has a valid IP Address (ip_ad) and IP Subnet Mask (ip_snm) configured.

Furtheron the LAN must be enabled, i.e. ip_lanstat=ENABLE and lanstat=ENABLE.

Error code:
`SNVS – Status, Not in Valid State, default router LAN port not configured`

Validation rule: check for correct OSI node configuration

The system shall not allow to set *ip_drlan* (IP Default Router LAN) as long as this LAN port is not assigned to OSI node 1(see ENT-FECOM-LAN).

Error code: `SNVS – Status, Not in Valid State, wrong IP configuration`

Validation rule: check that *raw_port* and *lv_port* are unequal

The **`ENT-ULS`** command is refused, if the two port parameters *raw_port* and *lv_port* are set to equal values.

Error code:
`IDNC – Input, Data Not Consistent, raw_port and lv_port must be unequal`

Validation rule: check of the maintenance condition

System must be in MCOND for change of *ip_ad_astn*.

Error code:
`SNVS – Status, Not in Valid State, Not in maintenance condition`

Validation rule: check of *xcsetup* for *ip_ad_astn* change

On a change of *ip_ad_astn* the system shall check that *xcsetup* = TRADITIONAL.

Error code: `SNVS – Status, Not in Valid State, xcsetup must be TRADITIONAL`

Validation rule: check of the maintenance condition for UNI IP address change

System must be in MCOND for change of *ip_ad_uni*.

Error code:
`SNVS – Status, Not in Valid State, Not in maintenance condition`

Validation rule: check of the maintenance condition for SC IP address and/or SC ID change.

System must be in MCOND for change of *sigPCIP_Address* and/or *sigPCID*.

Error code:
`SNVS – Status, Not in Valid State, Not in maintenance condition`

Validation rule: *sigPCIP_Address* and *sigPCID* must be supplied together.

The command is rejected if *sigPCIP_Address* is supplied, but *sigPCID* is not. It shall also be rejected if *sigPCID* is supplied, but *sigPCIP_Address* is not.

Error code:
`IDNC – Input, Data Not Consistent, sigPCIP_Address and sigPCID must be supplied together`

Validation rule: UNI IP address, SC IP Address and SCN IP address must be different.

The command is rejected if any two of *ip_ad_uni*, *sigPCIP_Address*, and *ip_ad_astn* have identical values.

Error code:

`IDNC – Input, Data Not Consistent, UNI IP address, sigPCIP address, and ASTN IP address must not be equal`

Validation rule: SC IP Address and RC IP address must be different

The command is rejected if *sigPCIP_Address* and *RoutingControlIPAddr* in **`ACT-L1ROUTING`** have identical values.

Error code:

`IDNC – Input, Data Not Consistent, sigPCIP address and RC IP address must not be equal`

Related TL1 messages

- "RTRV-ULS: Retrieve upper layer stack" (p. 3-3045)

ENT-ULS notification

Output format

The **ENT-ULS** notification has the following specific syntax:

ENT-ULS:::*spec_block*

The output parameters *sid*, *date*, *time*, and *ctag* are described in the REPT DBCHG notification.

Output parameters

1. **ip_ad_astn** – IP Address of ASTN
   This parameter is used to set the router ID for the SCN and the SNN node.
   Note: The provisioning of this IP address is ONLY ALLOWED ONCE.
   Type: Four dot separated numeric strings.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.

   The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – IP address for UNI
   This parameter is used to set the UNI node IP address.
   Type: Four dot separated numeric strings.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.
3. **sigPCIP_Address** – IP address for Signaling Controller (E-NNI node)
   This parameter is used to set the signaling controller IP address (E-NNI node address).
   Type: Four dot separated numeric strings.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.

   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
4. **sigPCID** – Identifier of the Signaling Controller (E-NNI node)
   This parameter is used to set the signaling controller identifier.
   Type: Four dot separated numeric strings.
   Range: Four times numeric string: 0 .. 255
5. **ip_drad** – IP Default Router Address
   This parameter is used to set or retrieve the IP default router address (IPv4) to which the host points to, needed for the T-TD functionality.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.

6. **`ip_drlan`** – IP Default Router LAN port AID
   This parameter specifies with which LAN port the default router is associated.
   Type: AID.
   Range: LanPortAID. The value lanall is not allowed.
   Default: 1-1-lan1.
7. **`lv_port`** – T-TD Length Value Port
   This parameter is used to set or retrieve the port number over which the T-TD function is accessible using the Length Value encoding method.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3081.
8. **`raw_port`** – T-TD Raw Mode Port
   This parameter is used to set or retrieve the port AID of number over which the T-TD function is accessible using raw-mode encoding.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3082.

Example output

The following is an example ENT-ULS notification in case the lv_port parameter was changed:

```
   LT-UNITE 00-11-01 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:ENT-ULS:::lv_port=7777″
;
```

ENT-ULSDCC-L3: Create a further OSI node

The command **ENT-ULSDCC-L3** is used to create a new OSI node. Up to 8 independent OSI instances can exist in a NE: Node-1 has a NSAP and the nodes 2-8 have each a Network Entity Title (NET). NSAP = NET + SEL Node-1 with AID osinode1 is created by start up of the UNITE NE as default. Node 2-8 can be created and configured with this command by entering an osinode AID which is not used until now, a NET area address and an IS-IS Level. Parameter in the SYSTEM field of the NET are predefined. Further parameter for configuration can be set like IS domain, RA prefix and RA destination node.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-ULSDCC-L3**:*tid*::*ctag*::*osinode*:*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 (osinode1) is default. Node 2-8 (osinode{2-8}) can be created and configured.
   Type: AID.
   Range: osinode{2-8}.

Spec block parameters

1. **l3afi** – NSAP/NET Authority and Format Identifier
   This parameter sets or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99
   Default: 39. This value identifies the iso_dcc format.
2. **l3idi** – NSAP/NET initial Domain Identifier

This parameter consists of two values:

- a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
- a 1 semi-octet (1-digit hexadecimal) IDI pad part. The parameter is used for set and retrieve values.

Type: Hexadecimal number.
Range: 0000 – FFFF
Default:
SONET: IDI = 840; IDI pad = F
SDH: IDI = 000; IDI pad = 0.

3. **`l3dfi`** – NSAP/NET DSP Format Identifier
This parameter (2-digit hex) NSAP is used to set or retrieve the format for the rest of the address.
Type: Hexadecimal number.
Range: 00 – FF.
Default: 80. This value represents the SONET DSP format.
4. **`l3org`** – NSAP/NET Organization Identifier
This parameter of NSAP is used to set or retrieve the NSP (allocated Company Code).
Type: Hexadecimal number.
Range: 000000 – FFFFFF.
Default: 000000.
5. **`l3res`** – NSAP/NET Reserved
This NSAP part currently has not been assigned a specific purpose.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.
6. **`l3rd`** – NSAP/NET Routing Domain
This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.
7. **`l3area`** – NSAP/NET Area Identifier
This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
Type: Hexadecimal number.
Range: 0000 – FFFF.
Default: 0000.

8. **l3isisnl** – Layer 3 IS Level
   This parameter indicates the network router type: ES-IS (level1, value L1), IS-IS (level2, value L2) and partition repair capability (value3).
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Alphanumeric string.
   Range:
   - L1: this means level1, IS-IS, only allowed when parameter l3radesnode=0
   - L12: this means level2, IS-IS which contains also level1, ES-IS
   - L12PRC : this means partition repair capable (PRC) and contains also level1 and level2.

   Default: L1.
9. **l3domain** – Layer3 IS Domain
   Per node in a NE, a domain group must be specified. This will be done with this parameter which can be set and retrieved. All nodes in the NE with the same domain-number are interconnected and can communicate without the need for inter-domain routing via RA-PREFIX.
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Alphanumeric string.
   Range:
   - 1 for osinode1
   - 1 – 2 for osinode2 to osinode8.

   Default: 1.
10. **l3raprefix** – Layer3 Reachable Address Prefix
   This value specifies an area address or a part of an area address representing an area address group. This parameter is needed when nodes in different routing domains (see *l3domain*) must communicate. The node must forward then CLNP PDUs to the RA-Destination-Node if the destination NSAPs of CLNP PDUs match with the PA-Prefix.
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Hexadecimal number.
   Range:
   - NONE,
   - a string of 1 to 13 bytes: first byte (most significant): 10 – 99; rest: 0000 – FFFF.

   The value represents an area address or a part of an area address beginning with AFI.
   Default: empty string.

11. **l3radesnode** – Layer3 Reachable Address Destination Node
    This parameter is in combination with parameter l3raprefix necessary. The value represents the destination node in the same NE. An internal connection over a domain-boundary is established between this node and the RA-Destination-Node, with IS-IS disabled over that connection.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: AID and digit value.
    Range:
    - 0, osinode{2-8} for osinode1
    - 0, osinode1 for osinode2 to osinode 8.
    Value 0 indicates that functionality of RA-Prefix is disabled.
    Default: 0.
12. **nsapthlev** – Number of OSI Nodes Threshold Level
    This parameter indicates the NSAP count threshold level for monitoring the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the *osinode* parameter is located. A threshold value of zero (0) indicates that NSAP threshold alarming is disabled. The parameter is omitted for the SCN node.
    Type: Numeric string.
    Range: 0 no alarming; 1 – 250threshold alarming
    Default: 0

Output format

If the **ENT-ULSDCC-L3** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command creates a new OSI node as specified in the command line. The command leads to a reset of the DCF controller and disconnection of management communication. The update is reported by means of **ENT-ULSDCC-L3** notification.

Example input/output

The following example shows the result of **ENT-ULSDCC-L3**:

```
ENT-ULSDCC-L3:NE-
      UNITE::4744843::osinode8:l3afi=39,l3idi=840F,l3dfi=80,l3org=100100,l3res=0000,l3rd=0020,l
```

```
   NE-UNITE 01-01-15 11:59:02
M  4744843 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **ENT-ULSDCC-L3** command.

Validation rule: OSI Node is already created

If the command is used with a value for parameter osinode which is already used for a creation of a OSI node, the command is rejected. This means the operation is rejected when the node ID is already in the created state.

Error code: `IEAE – Input, Entity Already Exists, osinode`

Validation rule: Address area value – RA-Prefix value

If a new RA-Prefix matches with the concerning part of the new area address the provisioning of the new NSAP area address must be rejected.

Error code:
`IDNV – Input, Data Not Valid, NSAP Area Address matches with RA-Prefix`

Validation rule: RA-prefix value – Address area value

If a RA-Prefix will be initiated and the value matches with the concerning part of the NSAP Area Address of this node the command must be rejected. The prefix should be specific enough, so that it matches not with NSAPs in both domains.

Error code:
`IDNV – Input, Data Not Valid, RA-Prefix matches with the NSAP Area Address`

Validation rule: *l3radesnode* value must be different from osinode

The command will be rejected when parameter *l3radesnode* is equal to the *osinode* parameter of the node.

Error code: `IDNV – Input, Data Not Valid, equal node IDs`

Validation rule: *l3radesnode* must be in the created state

If the destination node (*l3radesnode* parameter) is not in the created state, the command must be rejected.

Error code:
`SNVS – Status, Not in Valid State, l3radesnode not in created state`

Validation rule: consistency of *l3radesnode* with *l3raprefix*

If the value for the RA Destination Node (l3radesnode parameter) is equal to 0 while the value of the RA Prefix (l3raprefix parameter) is not NONE the command must be rejected.

Error code:

`IDNV – Input, Data Not Valid, l3prefix and l3radesnode inconsistent`

Validation rule: *l3radesnode* value and *l3isisnl* value

The l3isisnl parameter value L1 (Level1) is only allowed when the entered l3radesnode=0

Error code: `IDNV – Input, Data Not Valid, constraint l3radesnode`

Validation rule: only one single node in a separate domain supported

The command is rejected when l3domain has an other value than 1 and there is already an other osinode which has a value other than 1 for l3domain.

Error code:

`IDNV – Input, Data Not Valid, only a single node in separate domain supported`

Validation rule: consistency *l3domain* with *l3radesnode* for nodes 2 to 8

The command is rejected when l3domain for osinode2 to osinode8 has a value of 1 and l3radesnode is not equal to 0.

Error code:

`IDNV – Input, Data Not Valid, inconsistency l3domain with l3radesnode`

Related TL1 messages

- "RTRV-ULSDCC-L3: Retrieve layer 3 related parameters" (p. 3-3049)

ENT-ULSDCC-L3 notification

The command **ENT-ULSDCC-L3** effects a communication SW-reset (protocol stack) and the management connection is lost. Therefore it cannot be guaranteed that a notification could be returned as described below.

Output format

The ENT-ULSDCC-L3 notification has the following specific syntax:

ENT-ULSDCC-L3*::osinode:spec_block*

Output parameters

Command parameters

1. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 (osinode1) is default. Node 2-8 (osinode{2-8}) can be created and configured.
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. **l3afi** – NSAP/NET Authority and Format Identifier
   This parameter sets or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99
   Default: 39. This value identifies the iso_dcc format.
2. **l3idi** – NSAP/NET initial Domain Identifier
   This parameter consists of two values:
   - a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
   - a 1 semi-octet (1-digit hexadecimal) IDI pad part. The parameter is used for set and retrieve values.

   Type: Hexadecimal number.
   Range: 0000 – FFFF
   Default:
   SONET: IDI = 840; IDI pad = F
   SDH: IDI = 000; IDI pad = 0.
3. **l3dfi** – NSAP/NET DSP Format Identifier
   This parameter (2-digit hex) NSAP is used to set or retrieve the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 00 – FF.
   Default: 80. This value represents the SONET DSP format.

4. **`l3org`** – NSAP/NET Organization Identifier
   This parameter of NSAP is used to set or retrieve the NSP (allocated Company Code).
   Type: Hexadecimal number.
   Range: 000000 – FFFFFF.
   Default: 000000.
5. **`l3res`** – NSAP/NET Reserved
   This NSAP part currently has not been assigned a specific purpose.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
6. **`l3rd`** – NSAP/NET Routing Domain
   This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
7. **`l3area`** – NSAP/NET Area Identifier
   This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
8. **`l3isisnl`** – Layer 3 IS Level
   This parameter indicates the network router type: ES-IS (level1, value L1), IS-IS (level2, value L2) and partition repair capability (value3).
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Alphanumeric string.
   Range:
   - L1: this means level1, IS-IS, only allowed when parameter l3radesnode=0
   - L12: this means level2, IS-IS which contains also level1, ES-IS
   - L12PRC : this means partition repair capable (PRC) and contains also level1 and level2.

   Default: L1.
9. **`l3domain`** – Layer3 IS Domain
   Per node in a NE, a domain group must be specified. This will be done with this parameter which can be set and retrieved. All nodes in the NE with the same domain-number are interconnected and can communicate without the need for inter-domain routing via RA-PREFIX.
   Note: This parameter is not reported when output parameter *osinode* = scnnode.
   Type: Alphanumeric string.

Range:

- 1 for osinode1
- 1 – 2 for osinode2 to osinode8.

Default: 1.

10. **l3raprefix** – Layer3 Reachable Address Prefix
    This value specifies an area address or a part of an area address representing an area address group. This parameter is needed when nodes in different routing domains (see *l3domain*) must communicate. The node must forward then CLNP PDUs to the RA-Destination-Node if the destination NSAPs of CLNP PDUs match with the PA-Prefix.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Hexadecimal number.
    Range:
    - NONE,
    - a string of 1 to 13 bytes: first byte (most significant): 10 – 99; rest: 0000 – FFFF.

    The value represents an area address or a part of an area address beginning with AFI.
    Default: empty string.
11. **l3radesnode** – Layer3 Reachable Address Destination Node
    This parameter is in combination with parameter l3raprefix necessary. The value represents the destination node in the same NE. An internal connection over a domain-boundary is established between this node and the RA-Destination-Node, with IS-IS disabled over that connection.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: AID and digit value.
    Range:
    - 0, osinode{2-8} for osinode1
    - 0, osinode1 for osinode2 to osinode 8.

    Value 0 indicates that functionality of RA-Prefix is disabled.
    Default: 0.
12. **nsapthlev** – Number of OSI Nodes Threshold Level
    This parameter indicates the NSAP count threshold level for monitoring the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the *osinode* parameter is located. A threshold value of zero (0) indicates that NSAP threshold alarming is disabled. The parameter is omitted for the SCN node.
    Type: Numeric string.
    Range: 0 no alarming; 1 -250 threshold alarming
    Default: 0

Example output

```
   NE-UNITE 01-01-13 01:57:42
A  000 REPT DBCHG
”TIME=01-47-43,DATE=01-01-13:ENT-ULSDCC-L3::
osinode7:l3afi=39,l3idi=850F,l3dfi=80,l3org=100100,
l3res=0000,l3rd=0020,l3area=4711,l3isisnl=Ll2,
l3domain=2,l3raprefix=3984,l3radesnode=osinode1,
nsapthlev=0”
;
```

ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks

Among other Layer 4 covers the TARP functionality. The parameter of this command for 1675 LambdaUnite MSS NEs are only related to the TARP.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-ULSDCC-L4**:*tid*::*ctag*::*osinode*:[*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack. Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured.
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. **l4etdc** – TARP Data Cache
   This parameter is only valid for node-1. It is used to enable or disable TARP data cache.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE.
2. **l4etof** – TARP origination
   This parameter only valid for node-1. It is used to enable or disable TARP Origination function and is valid for all interfaces of a node.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: ENABLE (for osinode1), DISABLE (for osinode2-8, fixed).

3. **l4etpf** – TARP propagation
   This parameter is used to enable or disable TARP Propagation function for a node.
   Type: Alphanumeric value set.
   Range:
   - ENABLE (default)
   - DISABLE.
4. **l4etrf** – TARP responder
   This parameter is only valid for node-1. It is used to enable or disable TARP responder function
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE.

   Default: For node-1: ENABLE DISABLE (fixed for nodes 2-8).
5. **l4ajafi** – TARP Man. Adjacency: NET Authority and Format Identifier
   This parameter of NSAP is used to set or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99.
   Default: Value of authority and format identifier part of NET of node defined by parameter *osinode*.
6. **l4ajidi** – TARP Man. Adjacency: NET initial Domain Identifier
   This NSAP part consists two values:
   - a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
   - a 1 semi-octet (1-digit hexadecimal) IDI pad part.

   The parameter is used for set and retrieve values.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: Value of initial domain identifier part of NET of node defined by parameter *osinode*.
7. **l4ajdfi** – TARP Man. Adjacency: NET DSP Format Identifier
   The purpose of this 1 octet (2-digit hex) NSAP part is to set or retrieve the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 00 – FF.
   Default: Value of DSP format identifier part of NET of node defined by parameter *osinode*.

8. **l4ajorg** – TARP Man. Adjacency: NET Organization Identifier
   This parameter of NSAP is used to set or retrieve the NSP (allocated company code).
   Type: Hexadecimal number.
   Range: 000000 – FFFFFF.
   Default: Value of organization part of NET of node defined by parameter *osinode*.
9. **l4ajres** -TARP Man. Adjacency: NET Reserved
   This NSAP part currently has not been assigned a specific purpose.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: Value of reserved part of NET of node defined by parameter *osinode*.
10. **l4ajrd** – TARP Man. Adjacency: NET Routing Domain
    This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
    Type: Hexadecimal number.
    Range: 0000 – FFFF.
    Default: Value of routing domain part of NET of node defined by parameter *osinode*.
11. **l4ajarea** – TARP Man. Adjacency: NET Area Identifier
    This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
    Type: Hexadecimal number.
    Range: 0000 – FFFF.
    Default: Value of area part of NET of node defined by parameter *osinode*.
12. **l4ajsys** – TARP Man. Adjacency: NET System Identifier
    This system parameter of the NSAP is set by the Equipment Manufacturer. The value is fix and globally unique among all NEs. For values of NSAP/NET system parameter.
    Type: Hexadecimal number.
    Range: 000000000000 – FFFFFFFFFFFF.

Output format

If the **ENT-ULSDCC-L4** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The command sets respectively modifies layer 4 parameter for a node as specified in the spec_block of the command line. Result should be an update of the OSI stack.

Modification of any of the parameters *l4etof*, *l4etpf*, *l4etrf* leads to a reset of the DCF controller and disconnection of management communication.

The update is reported by means of **`ENT-ULSDCC-L4`** notification.

Up to two TARP manual adjacencies are possible for each node in an 1675 LambdaUnite MSS NE. One manual adjacency is added with each command request.

In order to update/modify an already provisioned manual adjacency, the following command sequence must be applied:

1. delete the manual adjacency to be modified by means of **`DLT-ULSDCC-L4`**
2. add the manual adjacency with the new values by means of **`ENT-ULSDCC-L4`**.

The sequence of disabling and enabling the TARP Data Cache (l4etdc=DISABLE – l4etdc=ENABLE) shall flush the TDC. If remote NEs are accessed via a gateway NE using the T-TD function, remote NEs may not be reachable after NSAP changes at the remote NEs. In this case flushing of the TARP Data Cache at the gateway NE by disabling and enabling it can be used to reestablish the accessibility of remote NEs.

Example input/output

The following example shows the result of **`ENT-ULSDCC-L4`**:

```
ENT-ULSDCC-L4:NE-
      UNITE::4744843::osinode1:l4ajafi=39,l4ajidi=840F,l4ajdfi=80,l4ajorg=100100,l4ajres=0000,l

   NE-UNITE 01-01-18 10:09:52
M  4744843 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ENT-ULSDCC-L4`** command.

Validation rule: TARP origination only valid for node-1

The parameter *l4etof* can only be set to ENABLE for osinode1. When trying to ENABLE *l4etof* for osinode2-8 the command should be rejected.

Error code: `IDRG – Input, Data Range Error, osinode`

Validation rule: TARP responder only valid for node-1

The parameter *l4etrf* can only be set to ENABLE for osinode1. When trying to ENABLE *l4etrf* for osinode2-8 the command should be rejected.

Error code: `IDRG – Input, Data Range Error, osinode`

Validation rule: TARP data cache enabling only valid for node-1

The parameter *l4etdc* can only be set to ENABLE or DISABLE for osinode1. When trying to set *l4etdc* for osinode2-8 the command should be rejected.

Error code: `IDRG – Input, Data Range Error, osinode`

Validation rule: TARP data cache change only if TARP origination enabled

The parameter *l4etdc* can only be changed while the TARP origination function is enabled (l4etof=ENABLE).

Error code:
`SNVS – Status, Not in Valid State, TARP origination function disabled`

Validation rule: Not allowed to disable TARP when T-TD is enabled

The parameters *l4etof*, *l4etpf*, *l4etrf* for osinode1 can not be disabled when T-TD is enabled (see "ED-NE: Edit NE" (p. 3-442) *tl1tcpipgw*). The command should be rejected in this case.

Error code: `SNVS – Status, Not in Valid State, T-TD enabled`

Validation rule: Can't disable TARP propagation while LAN TARP storm suppression is enabled

If it is tried to set l4etpf=DISABLE while lantlss=ENABLE (see RTRV-FECOM-LAN) the command is denied.

Error code:
`INVS – Status, Not in Valid State, TARP propagation can not be disabled`

Validation rule: *l4ajsys* must be supplied

The parameter *l4ajsys* must be supplied when manual adjacencies are configured, i.e. when any of the other NET parameters *l4ajafi, l4ajidi, l4ajdfi, l4ajorg, l4ajrd, l4ajarea* is present, otherwise the command should be rejected.

Error code: `IDNC – Input, Data Not Consistent`

Validation rule: *l4ajsys* per osinode must be different

Each manual adjacency for one osinode must have a different value for the parameter *l4ajsys*, i.e. all pairs of *osinode*/*l4ajsys* must be unique in the system.

Error code: `IDNV – Input, Data Not Valid, l4ajsys/osinode not unique`

Validation rule: no more than two TARP manual adjacencies per node

In case two TARP manual adjacencies are already created for a node, the creation of a third TARP manual adjacency for that node is denied:

Error code: `SROF – Status, Requested Operation Failed`

Validation rule: Provisioning of TARP manual adjacencies only if TARP enabled

The system shall deny an attempt to provision a TARP manual djacency if the TARP Origination and the TARP Propagation functions are disabled.

Error code: `SROF – Status, Requested Operation Failed`

Related TL1 messages

- "RTRV-ULSDCC-L4: Retrieve OSI layer 4 related parameters" (p. 3-3055)
- "DLT-ULSDCC-L4: Delete TARP manual adjacent NSAP address in layer 4 of the specified OSI stack" (p. 3-269)

ENT-ULSDCC-L4 notification

Output format

Syntax: The **ENT-ULSDCC-L4** notification has the following specific syntax:

ENT-ULSDCC-L4::*osinode*:*spec_block*

Output parameters

Command parameters

1. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack.
   Node-1 is created by start up of the 1675 LambdaUnite MSS NE as default. Node 2-8 can be created and configured.
   Type: AID.
   Range: osinode{1-8}.

Spec block parameters

1. **l4etdc** – TARP Data Cache
   This parameter is only valid for node-1. It is used to enable or disable TARP data cache.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE
2. **l4etof** – TARP origination
   This parameter only valid for node-1. It is used to enable or disable TARP Origination function and is valid for all interfaces of a node.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: ENABLE (for osinode1), DISABLE (for osinode2-8, fixed).
3. **l4etpf** – TARP propagation
   This parameter is used to enable or disable TARP Propagation function for a node.
   Type: Alphanumeric value set.
   Range:
   - ENABLE (default)
   - DISABLE.
4. **l4etrf** – TARP responder
   This parameter is only valid for node-1. It is used to enable or disable TARP responder function
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE.

   Default: For node-1: ENABLE DISABLE (fixed for nodes 2-8).

5. **l4ajafi** – TARP Man. Adjacency: NET Authority and Format Identifier
   This parameter of NSAP is used to set or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99.
   Default: Value of authority and format identifier part of NET of node defined by parameter *osinode*.
6. **l4ajidi** – TARP Man. Adjacency: NET initial Domain Identifier
   This NSAP part consists two values:
   - a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
   - a 1 semi-octet (1-digit hexadecimal) IDI pad part.

   The parameter is used for set and retrieve values.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: Value of initial domain identifier part of NET of node defined by parameter *osinode*.
7. **l4ajdfi** – TARP Man. Adjacency: NET DSP Format Identifier
   The purpose of this 1 octet (2-digit hex) NSAP part is to set or retrieve the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 00 – FF.
   Default: Value of DSP format identifier part of NET of node defined by parameter *osinode*.
8. **l4ajorg** – TARP Man. Adjacency: NET Organization Identifier
   This parameter of NSAP is used to set or retrieve the NSP (allocated company code).
   Type: Hexadecimal number.
   Range: 000000 – FFFFFF.
   Default: Value of organization part of NET of node defined by parameter *osinode*.
9. **l4ajres** -TARP Man. Adjacency: NET Reserved
   This NSAP part currently has not been assigned a specific purpose.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: Value of reserved part of NET of node defined by parameter *osinode*.
10. **l4ajrd** – TARP Man. Adjacency: NET Routing Domain
   This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: Value of routing domain part of NET of node defined by parameter *osinode*.

11. **`l4ajarea`** – TARP Man. Adjacency: NET Area Identifier
    This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
    Type: Hexadecimal number.
    Range: 0000 – FFFF.
    Default: Value of area part of NET of node defined by parameter *osinode*.
12. **`l4ajsys`** – TARP Man. Adjacency: NET System Identifier
    This system parameter of the NSAP is set by the Equipment Manufacturer. The value is fix and globally unique among all NEs. For values of NSAP/NET system parameter.
    Type: Hexadecimal number.
    Range: 000000000000 – FFFFFFFFFFFF.

Example output

```
   NE-UNITE 01-01-19 04:29:01
A  000 REPT DBCHG
   ”TIME=04-29-00,DATE=01-01-19:ENT-ULSDCC-
      L4::osinode1:l4etof=DISABLE,l4etpf=ENABLE,l4etrf=DISABLE,l4ajafi=39,l4ajidi=0,l4ajdfi=
;
```

ENT-USER-SECU: Enter user security

The **ENT-USER-SECU** command is used by an administrator to enter a new user and security parameters associated with that user.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: No.

Input format

Command syntax: **ENT-USER-SECU**:*tid*:*uid*:*ctag*::*pid*,,[*upc*]:[*spec_block*];

Input parameters

Note: The new setting becomes operational only after a new login.

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **uid** – User ID
   This parameter specifies the new user ID (*uid*) name.
   Type: Alphanumeric string (case-sensitive)
   Range: 1 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **pid** – Password ID
   This parameter specifies the password ID for the *uid*.
   This required parameter is the confidential password authenticator for the given *uid*.
   Password strings are transmitted in non-encrypted form in the **ACT-USER** command; are encrypted when stored in the NE; and are never transmitted from the NE.
   Type: character string (case-sensitive)
   Range: 6 to 10 legal characters. Valid passwords consist of at least two

non-alphabetic characters with at least one special character. The special character can be one of the required non-alpha characters. For example, TEST1+ is a valid password. The first character of a password must be a letter.
Default: For the two superusers the default password is: UNITE+01 for User ID LUC01 and UNITE+02 for User ID LUC02. For other users: not applicable.

5. **upc** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation "&" is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also "List of user privilege codes" (p. 15-1).

Spec block parameters

1. **page** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90
2. **screen** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string. An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG
     Data Base Change notifications: changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes).
     File Transfer In-progress notifications: The user will receive **REPT DBCHG** and REPT EVT FXFR messages.
     The value DBCHG shall always be used together with the value STCHG.
   - STCHG

State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **REPT DBCHG** messages.

The value STCHG shall always be used together with the value DBCHG.

- PSCHG

  Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.

- ALARMS

  Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.

- ONNS

  ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.

- SECURITY: Security event notifications

- ALL

  All notifications.

  All Notifications: This value can be used in input direction as a short-hand notation for DBCHG&STCHG&PSCHG&ALARMS&ONNS&SECURITY

- OWN

  Responses to own commands. The user will receive no other notifications.

- NA

  Not applicable. For users with uidclass=MEMADMIN only this value is allowed. The notification behavior is controlled in this case by the user class. If the value NA is applied for users of other user classes, then the value has the same effect as DBCHG.

- Default: OWN

  If more than one value is set the values will be concatenated by '&'. Example: DBCHG&STCHG&ALARMS

Restriction: When *uidclass* is set to MEMADMIN, *screen* must be set to NA.

3. **tmout** – Inactivity Timeout Period
   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.
   Type: Integer.
   Range: 0 – 999 [minutes]
   Default: 30

4. **uidclass** – UID Class
   This is the user class of a *uid*.
   Type: Alphanumeric value set.

Range:

- OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
- MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **RTRV-AO**.

Default: OTHER.

Restriction: When *uidclass* is set to MEMADMIN, *screen* must be set to NA.

Output format

After receiving the **ENT-USER-SECU** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

This command creates the NE’s user data. A new user *uid* (User ID) will be established. The values of *pid* (Password ID), *upc* (User Privilege Code List), *alw_login* (Allow Login), *screen* (Message Screening), *page* (Password Aging Interval), *tmout* (Inactivity Timeout Period) will be assigned to the new user. If optional values are not specified the default values will be used.

This update is reported via the corresponding ENT-USER-SECU notification – see “ENT-USER-SECU notification” (p. 3-1975)section.

The execution of ENT-USER-SECU command is entered in the Security Information Log.

The NE will always have two preinstalled users with full privileges in all functional categories. These two users will be referred to as super-users. The two super-users are also Security Administrators because they have the S5 privilege code. The NE will not allow adding additional super-user type users. It will not be possible to remove either of the original super-user login IDs. An existing login ID (*uid*) except the super-user login IDs cannot be renamed. It must be deleted and re-entered with new *uid* value.

Example input/output

The following example shows the successful completion of the **ENT-USER-SECU** command by the NE:

ENT-USER-SECU:LT-UNITE-123:nbguser1:123456::pass12+,,P3&M3:screen;=DBCHG&STCHG;&PSCHG;,page=345,tmout=360;

```
   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Entering user's own or existing uid.

If the NE receives an ENT-USER-SECU command from a user trying to enter his or her own *uid* or an existing *uid*.

Error code: `IDNV – Input, Data Not Valid, duplicate UID`

Validation rule: Maximum number of *uids* reached

If the NE receives an ENT-USER-SECU command when it already has reached its maximum number of defined User IDs.

Error code: `SLEM – Status, List Exceeds Maximum UIDs`

Validation rule: Invalid *upc*

If the NE receives an ENT-USER-SECU command with a upc=P5&M5&T5&S5&PM5 or a upc=S0.

Error code: `IDNV – Input, Data Not Valid, invalid UPC`

Validation rule: Compatibility screen, uidclass parameter

If the NE receives an ENT-USER-SECU command with *uidclass* = MEMADMIN and *screen* !=NA it rejects the command with

Error code:
`IDNV – Input, Data Not Valid, incompatible` *`screen, uidclass`* `parameters`

Related TL1 messages

- "DLT-USER-SECU: Delete user security" (p. 3-272)
- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

ENT-USER-SECU notification

Output format

The **ENT-USER-SECU** notification has the following specific syntax:

ENT-USER-SECU:*uid*:,,[*upc*]:*spec_block*

Output parameters

Command parameters

1. **uid** – User Identifier
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of *uid* values for the **ACT-USER** command are not allowed. Valid *uid* values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 5 – 10 characters
   The *uid* ALL is not allowed in any combination of upper or lower cases.
   Default: not applicable
2. **upc** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation "&" is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also "List of user privilege codes" (p. 15-1).

Spec block parameters

1. **alw_login** – Allow Login
   This parameter displays the enable/disable state of a User ID.
   A User ID can be automatically disabled by the system when the time specified by parameter *usrage* (see "RTRV-NE-SECU: Retrieve NE security" (p. 3-2552)) expires or via the **ED-USER-SECU** command. A User ID can be enabled via the **ED-USER-SECU** command.
   If a User ID is disabled, logins with that User ID are denied.
   Type: Alphanumeric value set.

Range:

- YES: User ID is enabled
- NO: User ID is disabled

Default: YES

2. **page** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90
3. **screen** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string. An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG

     Data Base Change notifications: changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes).

     File Transfer In-progress notifications: The user will receive **REPT DBCHG** and REPT EVT FXFR messages.
   - STCHG

     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **REPT DBCHG** messages.
   - PSCHG

     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS

     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
   - ONNS

     ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN.

     The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
   - SECURITY: Security event notifications
   - OWN

     Responses to own commands. The user will receive no other notifications.

     Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.

- NA

  Not applicable. For users with uidclass=MEMADMIN only this value is allowed. The notification behavior is controlled in this case by the user class. If the value NA is applied for users of other user classes, then the value has the same effect as DBCHG.
- Default: OWN

  If more than one value is set the values will be concatenated by ’&’. Example: DBCHG&STCHG&ALARMS

4. **`tmout`** – Inactivity Timeout Period
   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.
   Type: Integer.
   Range: 0 – 999 [minutes]
   Default: 30
5. **`uidclass`** – UID Class
   This is the user class of a *uid*.
   Type: Alphanumeric value set.
   Range:
   - OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
   - MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **`RTRV-AO`**.

   Default: OTHER.
   Restriction: When *uidclass* is set to MEMADMIN, *screen* must be set to NA.

Example output

The example shows a normal notification after an ENT-USER-SECU Command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:ENT-USER-
      SECU:nbguser1:,,P3&M3:screen=DBCHG&STCHG&PSCHG,page=345,tmout=360,uidclass=OTHER,alw_l
;
```

ENT-VC3 notification

Output format

Syntax: **`ENT-VC3`**:*aid*::*spec_block*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Description: AID of the SDH tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **`crslpbkstat`** – Cross-connect loopback status
   State of cross- connect loopback.
   Type: Alphanumeric value set.
   Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
4. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
5. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.

6. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
7. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **`hptimmon`** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`loca`** – Ring A-node TID
   This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
   loca is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in “RTRV-HDR: Retrieve header” (p. 3-2417) description.
   Default: empty string: ″″. This is displayed as \″\″ in output direction.
10. **`locz`** – Ring Z-node TID
    This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.

locz is introduced specifically for use related to ONNS application.
Type: TID or empty-string.
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: "". This is displayed as \"\" in output direction.

11. **neppm** – Near-End Path PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
12. **pjepm** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
13. **pmhpfeses** – Performance Monitoring Path Far-End SES Threshold
    This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.
14. **pmhpneses** – Performance Monitoring Path Near-End SES Threshold
    This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.
15. **ptca** – Path TCA Profile Pointer
    Type: Alphanumeric string.
    Range: string with up to 24 characters.
    Default: DEFAULT0.
16. **ptrcrfmt** – Path Trace Read Format (J1)
    The read format of J1, the incoming path trace (ptrc).
    Type: numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16 byte specific string.
- 64: for 64 byte string.
  Default: 2.

17. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in BURST or POISSON mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON
18. **`substr`** – Substructured
    This parameter indicates if the tributary is substructured.
    Substructure for LO cross-connections:
    A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.
    The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).
    A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **`ED-SUBS`** command.
    Substructure for conversion cross-connections:
    A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.
    The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.
    A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.
    Type: Alphanumeric value set.
    Range:
    - YES: Automatically sub structured for LO cross-connections
    - MANSUB: Manually sub structured for LO cross-connections
    - CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
    - NO (default): Not sub structured, not used for conversion cross-connections
19. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (also known as the implicit reset method)
- TR-RTR (also known as the explicit reset method).
  Type: Alphanumeric value set.
  Range: TR, TRRTR (default).

20. **`tmonmd`** – Tributary Monitoring Mode
    This parameter indicates the mode of the tributary.
    Type: Alphanumeric value set.
    Range:
    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.
21. **`tribsdhpn`** – Tributary Alarm Severity Assignment Profile
    This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: string with a maximum of 24 characters.
    Default: DEFAULT.

Example output

```
   UNITE-NE 00-10-14 14:59:09
A  592 REPT DBCHG
   ″TIME=14-59-09,DATE=00-10-14:ENT-VC3:1-1-#-#-3-1-1::
      tmonmd=MON,tribsdhpn=Default,...,pmhpneses=30,pmhpfeses=30″
;
```

ENT-VCG – Enter virtual concatenation group

The **ENT-VCG** command provisions a virtual concatenation group of an ES64U together with all VCGTRIBs belonging to it.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Priority: 1

Abortable: No.

Input format

ENT-VCG-*modifier:tid:aid:ctag:::spec_block;*

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the virtual concatenation group of the unit. Implicit, it indicates the transmission rate for which VCG related data will be retrieved.
   The parameter *modifier* must be given if a virtual concatenation group on a GE10PL1 or ES64U pack shall be addressed. Otherwise this parameter has to be omitted.
   Type: Alphanumeric value set.
   Range:
   - 1GEPL representing a 1 GE port on the GE10PL1 pack
   - 10GEPL representing a 10 GE port on the GE10PL1 pack
   - ES64U representing a VCG port at an ES64U subsystem

   Default: None.
   The command is only allowed for *modifier* = ES64U.
2. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   The access identifier specifies the virtual concatenation group (VCG) port number to be created.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values
   - **vall** for Simple_VCGAID
   - **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID

- **all** for Simple_ShelfAID and
- **all** for BayAID

are excluded.
Default: None.

4. **ctag** – Correlation Tag
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **inputsig** – Tributary Input Signal Rate List
This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure depends on interface standard specified for this VCG by istd . For definition of *istd* refer to "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078) Output parameters.
If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the possible tributary structures depends on interface standard specified for this VCG by istd and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to RTRV-VCG Output Parameters.
Type: numeric string of the form NX
    - N specifies the count of container type X
    - X specifies the container type (or envelope).

    Range:
    - Values: N = 7, X = 4; Modifier=omitted; Interface Standard=SDH; Virtual Concatenation=not applicable; Represented Signal Structure: 7 times VC4
    - Values: N = 7, X = 4; Modifier=1GEPL; Interface Standard=SDH; Virtual Concatenation=ENABLE; Represented Signal Structure: 7 times VC4
    - Values: N = 64, X = 4; Modifier=10GEPL; Interface Standard=SDH; Virtual Concatenation=ENABLE; Represented Signal Structure: 64 times VC4
    - Values: N = 1, X = c64; Modifier=10GEPL; Interface Standard=SDH; Virtual Concatenation=DISABLE; Represented Signal Structure: 1 times VC4-64C
    - Values: N = 21, X = 1; Modifier=omitted; Interface Standard=SONET; Virtual Concatenation=not applicable; Represented Signal Structure: 21 times STS-1
    - Values: N = 21, X = 1; Modifier=1GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 21 times STS-1
    - Values: N = 7, X = 3; Modifier=1GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 7 times STS-3C
    - Values: N = 192, X = 1; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 921 times STS-1

- Values: N = 64, X = 3; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 64 times STS-3C
- Values: N = 1, X = 192; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=DISABLE; Represented Signal Structure: 1 times STS-192C

For *modifier* = ES64U, the value combination of “N” and “X” is done according to the following rules:

- *istd* = SONET
- X = 1 for STS-1
- N $\in$ {1, 2, …, 192}

For *modifier* = ES64U, the parameter *inputsig* is visible for retrieval, but it cannot be modified once the VCG is established.

Default:

- modifier = omitted, Interface Standard=SDH, Default Value = 74
- modifier = 1GEPL, Interface Standard=SDH, Default Value = 74
- modifier = 10GEPL, Interface Standard=SDH, Default Value = 644
- modifier = omitted, Interface Standard=SONET, Default Value = 211
- modifier = 1GEPL, Interface Standard=SONET, Default Value = 211
- modifier = 10GEPL, Interface Standard=SONET, Default Value = 1921
- modifier = ES64U, Interface Standard=SONET, Default Value not applicable

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, i.e. VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception: For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.

For *modifier* = ES64U, this parameter will always be defined by the **`ENT-VCG`** command.

2. **`vcat`** – Virtual Concatenation

The parameter *vcat* specifies whether the Ethernet signal is mapped to a number of virtually concatenated tributaries (value = ENABLE) or into a contiguously concatenated tributaries (value = DISABLE). In the latter case, LCAS related parameters become insignificant (values set to default, values cannot be changed). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *vcat* is not supported, then the parameter is omitted.

Type: Alphanumeric value set.

Range:

- For *modifier* = ES64U: ENABLE.
- Otherwise: ENABLE, DISABLE.

For *modifier* = ES64U, the parameter *vcat* is visible for retrieval, but cannot be modified once the VCG is established.

Default: Enable.

Output format

If the network element fully complies with the **ENT-VCG** request, the following normal completion response shall be returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **ENT-VCG** command generates a **REPT DBCHG** message for the creation of the VCG together with a **REPT DBCHG** message for the creation of the VCG tributaries (**ENT-VCGTRIB** notification).

The command establishes a virtual concatenation group together with the virtual concatenation group tributaries belonging to it according to the parameters given with this command. The virtual concatenation group tributary number range is from 1 up to the number of tributaries given by the parameter *inputsig*.

The modification is reported by means of an **ENT-VCG** notification and grouped notifications for VCG tributaries.

Group notification for VCG tributaries

For efficiency reasons the notifications for VCG tributaries shall be sent out as a single notification (so-called group notification) using a range over the AID which depends on the used mapping of the associated VCG port, i.e. the *Simple_VCGNumAID* field in the *VCGNumAID* is set to

- 1&&-***<N>*** for sequential range
- 1&-4…&-***<N>*** for non-sequential range

where ***<N>*** is the maximal legal VCG tributary number.

Example:

The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping)

1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping)

Example input/output

```
ENT-VCG-ES64U:UNITE-NE:1-1-#-#-22-v9:123456:::
inputsig=121,vcat=ENABLE;

  LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:ENT-VCG-ES64U:1-1-#-#-22-v9::\
    enetpn=DEFAULT,eportClass=TRADITIONAL,eportControl=NORMAL,\
    htmr=0,inputsig=121,istd=SONET,lcasmd=DISABLE,plcrthr=12,\
    plctthr=12,unequiposig=121,vcat=ENABLE,vmode=NMON,wtr=0,\
    clientNodeID=0.0.0.0,networkTNAType=IPV4,\
    networkTNAAddress=0.0.0.0.0,localLPID=0,\
    networkTNATypeVCAT=IPV4,networkTNAAddrVCAT=0.0.0.0.0,\
    localLPIDVCAT=0"
;

  LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:ENT-VCGTRIB-STS:1-1-#-#-22-v9-1&&-12::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-3,\
   hptimais=DISABLE,exphptrc=890000000000000000000000000000000,\
   hptimmon=DISABLE,inmbs=DISABLE,inmbst=NAV,outmbs=DISABLE,\
   outmbst=NAV,ptpn=DEFAULT,ptrcrfmt=64,ptrcwfmt=64,\
   sdsfmode=BURST,tmonmd=NMON,trc=00,tslmmon=ENABLE"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **INH-MSG** command.

Validation rule: Command only allowed for VCG on ES64U circuit pack

If the command is not applied to a VCG on a ES64U pack, then the command is denied.

Error code:

`IENE – Input, Entity Not Exists, Command only allowed for a ES64U pack`

Validation rule: Command not allowed for ES64U with switch capacity 0

If the switch capacity *es64ucap* for a ES64U pack is 0, then the command is denied.

Error code:

`IENE – Input, Entity Not Exists, Command not allowed for an inactive ES64U pack`

Validation rule: Command not allowed for ES64U protecting pack

If the command is applied to an ES64U protecting pack in an equipment protection, then the command is denied.

Error code:
`IENE – Input, Entity Not Exists, Command not allowed for a protecting ES64U pack`

Validation rule: Sufficient backplane bandwidth available

Check that sufficient backplane bandwidth is available before a new VCG is established. In general the number of allocated backplane bandwidth must be less than the maximum number of supported STS-1 equivalents, 192 for the first and 192 for the second segment. For contiguous concatenated tributaries it must be checked if the necessary coherent backplane bandwidth in available due to backplane capacity fragmentation issues.

Error code:
`SSRE – Status, System Resources Exceeded, no sufficient backplane bandwidth available`

Validation rule: VCG already exists

Check if the VCG number is already used.

Error code: `IDNV – Input, Data Not Valid, VCG already exists`

Validation rule: Parameter inputsig is mandatory

Check if the parameter *inputsig* is given as input.

Error code: `IDNV – Input, Data Not Valid, Parameter inputsig is mandatory`

Validation rule: Parameter vcat is mandatory

Check if the parameter *vcat* is given as input.

Error code: `IDNV – Input, Data Not Valid, Parameter vcat is mandatory`

Related TL1 messages

- "DLT-VCG – Enter virtual concatenation group" (p. 3-277)
- "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
- "RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group" (p. 3-2819)
- "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)

ENT-VCG notification

Output format

Syntax: The **ENT-VCG** notification has the following specific syntax:

ENT-VCG*[-modifier]:aid::spec_block;*

Output parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the virtual concatenation group of the unit. Implicit, for the 1GEPL, 10GEPL application, it indicates the transmission rate for which VCG related data will be notified.
   For the ES64U application, the transmission rate is defined by the user with the **ENT-VCG** command and input parameter *inputsig*.
   The parameter *modifier* must be given if a virtual concatenation group on a GE10PL1 or a ES64Upack is notified. Otherwise this parameter has to be omitted.
   Type: Alphanumeric value set.

   Range:

   - 1GEPL representing a 1 GE port on the GE10PL1 pack
   - 10GEPL representing a 10 GE port on the GE10PL1 pack
   - ES64U representing a VCG port at an ES64U subsystem

   Default: None.
2. **aid** – Access Identifier
   The access identifier specifies the virtual concatenation group (VCG) to be operated on.
   Type: AID.

   Range: VCGAID as defined in AID structure. The values

   - **vall** for Simple_VCGAID
   - **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID
   - **all** for Simple_ShelfAID and
   - **all** for BayAID

   is excluded.

Spec block parameters

1. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *eportClass* is set to a value supporting it.
This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other eport classes in future.
This parameter is supported by ports of *eportClass* UNI.
For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

2. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer WAN port or a network WAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
   If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER, for a customer WAN port. NETWORK, for a network WAN port.
   Default: NETWORK.
3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
   If the parameter *dftpri* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (i.e. from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
   If the parameter *dftvlan* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - 1, 2, 3, … 4092, 4093
   - UNAVAILABLE (default).
   To express the state UNAVAILABLE system internally the invalid VLAN ID 4094 is used.

5. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Ethernet port and VCG port have the same ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
6. **`enettca`** – Ethernet TCA Profile Name
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for the VCG.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   If the parameter *enettca* is not supported, then the parameter is omitted.
   Type: Alphanumeric string.
   Range: 1 … 24 characters length.
   Default: DEFAULT0.
7. **`eportClass`** – Ethernet Port Class
   This identifies the Ethernet port class of the AID. *eportClass* is used to specify Ethernet ports for use by NN application.
   This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
   Type: Alphanumeric value set.

   Range: Depends on the *modifier*:

   - if *modifier* is missing or is 1GEPL, then the value range is: TRADITIONAL, EDGE, UNI
   - if *modifier* is 10GEPL, then the value range is: TRADITIONAL, EDGE
   - if *modifier* is ES64U, then the value range is: TRADITIONAL

   Default: Depends on system parameters Cross Connect Setup (*xcsetup*) and Port Class Default (*portclassdefault*):

   - if *xcsetup* is TRADITIONAL, then the default value is TRADITIONAL
   - if *xcsetup* is SWISS, then the value of *portclassdefault* is used as default value.
   - if *modifier* is ES64U, then the value range is: TRADITIONAL
8. **`eportControl`** – Ethernet Port Control
   This identifies the administrative mode of Ethernet ports of *eportlass* = EDGE or UNI. For Ethernet ports of other Ethernet port class this parameter is not relevant. *eportControl* is introduced specifically for use by NN application.
   This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
   Type: Alphanumeric value set.
   Range: NORMAL (default), MAINT, LOCKOUT .

9. **`htmr`** – Hold-Off Time
   The parameter *htmr* specifies the hold off time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it goes to state faulty then this tributary is taken out of transmission by LCAS after the hold off time. This time is configured on group level, i.e. all tributaries have the same value for hold-off time. This parameter only affects NE behavior if lcasmd=ENABLE.
   This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
   Type: numeric value set.
   Range: 0 (default), 100, 200, …, 9900, 10000 [milliseconds].
10. **`inputsig`** – Tributary Input Signal Rate List
   This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure depends on interface standard specified for this VCG by *istd* . For definition of *istd* refer to the “Output parameters” section in “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).
   If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the possible tributary structures depends on interface standard specified for this VCG by istd and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to the “Output parameters” section in “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078)..
   Type: numeric string of the form ***NX***
   - ***N*** specifies the count of container type ***X***
   - ***X*** specifies the container type (or envelope).

   Range:

Values	Modifier	Interface standard	Virtual concatenation	Represented signal structure
N = 7, ***X*** = 4	omitted	SDH	not applicable	7 × VC4
N = 7, ***X*** = 4	1GEPL	SDH	ENABLE	7 × VC4
N = 64, ***X*** = 4	10GEPL	SDH	ENABLE	64 × VC4
N = 1, ***X*** = c64	10GEPL	SDH	DISABLE	1 × VC4-64C
N = 21, ***X*** = 1	omitted	SONET	not applicable	21 × STS-1
N = 21, ***X*** = 1	1GEPL	SONET	ENABLE	21 × STS-1
N = 7, ***X*** = 3	1GEPL	SONET	ENABLE	7 × STS-3C
N = 192, ***X*** = 1	10GEPL	SONET	ENABLE	192 × STS-1
N = 64, ***X*** = 3	10GEPL	SONET	ENABLE	64 × STS-3C
N = 1, ***X*** = 192	10GEPL	SONET	DISABLE	1 × STS-192C

Range: Depends on the *modifier*:

- if *modifier* is ES64U, then the value combination of ″N″ and ″X″ is according the following rules:
  istd=SONET
  X = 1 (for STS-1)
  N $\in$ {1, 2, … 192}
- if *modifier* is ES64U, then the parameter *inputsig* is visible for retrieval, but cannot be modified once the VCG is established.

Default:

- modifier = omitted, Interface Standard=SDH, Default Value = 74
- modifier = 1GEPL, Interface Standard=SDH, Default Value = 74
- modifier = 10GEPL, Interface Standard=SDH, Default Value = 644
- modifier = omitted, Interface Standard=SONET, Default Value = 211
- modifier = 1GEPL, Interface Standard=SONET, Default Value = 211
- modifier = 10GEPL, Interface Standard=SONET, Default Value = 1921

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, i.e. VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception:
For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.
For *modifier* = ES64U this parameter is always defined by the **`ENT-VCG`** command.

11. **`istd`** – Interface Standard
This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
For *modifier* = 1GEPL, 10GEPL or ES64U the parameter is visible for retrieval, but cannot be modified.
Type: Alphanumeric value set.
Range:
   - SDH
   - SONET.

   Default: *modifier* 1GEPL, 10GEPL or ES64U: The value is derived from the circuit pack parameter *istd*. no *modifier*. As derived from parameter *istddflt* specified at configuration management, section System RTRV-EQPT.

12. **`lcasmd`** – LCAS Mode
This parameter specifies the operation mode of the Link Capacity Adjustment Scheme for this VCG. This value is bidirectional.
This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
Type: Alphanumeric value set.
Range:
   - DISABLE
   - ENABLE (default).

Default:

- if *modifier* = ES64U: then DISABLE
- if other or no *modifier*: then ENABLE

13. **localLPID** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like, for example,*networkTNAAddress*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *localLPID* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
    Type: numeric value set.
    Range: 0 to 4294967295
    Default: 0

14. **localLPIDVCAT** – Local Logical Port Identity on VCAT layer
    The *localLPIDVCAT* provides a unique logical identification of a single VCG port with an *eportClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port, the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like, for example, *networkTNAAddrVCAT*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *localLPIDVCAT* is used by ONNS for switching network connections on VCAT layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    Type: numeric value set.
    Range: 0 to 4294967295.
    Default: 0.

15. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAType", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *networkTNAType* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
    Type: Alphanumeric value set.
    Range: IPV4, IPV6, or NSAP.
    Default: IPV4.
16. **`networkTNATypeVCAT`** – Network TNA Type on VCAT level
    The *networkTNATypeVCAT* identifies the type of the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *networkTNATypeVCAT* is used by ONNS for switching network connections on VCAT layer.
    This parameter is supported by ports of eportClass UNI.
    This parameter is supported by ports of eportClass EDGE.
    This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
    Type: Alphanumeric value set.
    Range: IPV4, IPV6, or NSAP.
    Default: IPV4.
17. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given NetworkTNAType. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAAddress", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic

name seems reasonable.
The parameter *networkTNAAddress* is used by ONNS for switching network connections on SONET/SDH layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
Type:
networkTNAType = IPV4: Four dot separated numeric string (represents 4 bytes)
networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 bytes)
networkTNAType = NSAP: Hexadecimal number (represents 20 Bytes)
Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Examples:
IPV4: 129.12.3.178
IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF
Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000

18. **`networkTNAAddrVCAT`** – Network TNA Address on VCAT level
The *networkTNAAddrVCAT* identifies the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNATypeVCAT*. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
The parameter *networkTNAAddrVCAT* is used by ONNS for switching network connections on VCAT layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
Type:
networkTNAType = IPV4: Four dot separated numeric string (represents 4 bytes)
networkTNAType = IPV6: Eight dash separated hexadecimal number (represents 16 bytes)
networkTNAType = NSAP: Hexadecimal number (represents 20 Bytes)

Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000

19. **plcrthr** – Threshold for Partial Loss of Capacity (Receive)
The parameter *plcrthr* specifies the acceptable limit of valid VCG members in receive direction (refer to the *outmbs* parameter of all related VCGTRIB entities) for which the PLCR alarm will not yet be raised. The PLCR alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plcrthr* parameter – 1)] (if *plcrthr* ≥ 2). The PLCR alarm shall never be raised if *plcrthr* is set to 1.
The PLCR alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plcrthr* parameter (or if the number of those VCG members is 0 – in this case the TLCR alarm will be raised instead). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plcrthr* is not supported, then the parameter is omitted.
This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
Type: numeric value set.
Range: 1, 2, 3, …, 192.
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192

20. **plctthr** – Threshold for Partial Loss of Capacity (Transmit)
The parameter *plctthr* specifies the acceptable limit of valid VCG members in transmit direction (refer to the *inmbs* parameter of all related VCGTRIB entities) for which the PLCT alarm will not yet be raised. The PLCT alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plctthr* parameter – 1)] (if *plctthr* ≥ 2). The PLCR alarm shall never be raised if *plctthr* is set to 1.
The PLCT alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plctthr* parameter (or if the number of those VCG members is 0 – in this case the TLCT alarm will be raised instead).

This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plctthr* is not supported, then the parameter is omitted.
This parameter is visible with default values for retrieval, but cannot be modified if the input parameter *modifier* = ES64U.
Type: numeric value set.
Range: 1, 2, 3, ..., 192.
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192

21. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, *modifier* = 10GEPL or *modifier* = ES64U.
    If the parameter *stdis* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: DISABLE, STP is disabled, i.e. the port is not used to build the spanning tree. ENABLE (default), STP is enabled, i.e. the port is used to build the spanning tree.
    As VCGs are by default network ports the default value is to enable STP.
22. **`stpcost`** – Spanning Tree Path Cost
    The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL,*modifier* = 10GEPL or *modifier* = ES64U
    If the parameter *stpcost* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 to 200000000.
    Default: 20000.
    The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

23. **`stpp`** – Spanning Tree Port Priority
    The parameter *stpp* specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database. This parameter is not supported if the input parameter: *modifier* = ES64U.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter *stpp* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.
    Default: 128.
24. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.
    Two thresholding methods are applicable:
    - TR-only (also known as the implicit reset method)
    - TR-RTR (also known as the explicit reset method)

    This parameter is not supported if the input parameter: *modifier* = ES64U. If the parameter *tcamd* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: TR, TRRTR.
    Default: TRRTR in SDH mode; TR in SONET mode.
25. **`unequiposig`** – Unequipped Tributary Output Signal Rate List
    The parameter *unequiposig* specifies the tributary structure and the capacity of the VCG port in output direction. As long as there is no cross-connection applied the specified tributaries are occupied by an unequipped signal. The parameter will implicitly give the maximum capacity of the VCG port, too.
    If the input parameter *modifier* is omitted, then the tributary structure depends on interface standard specified for this VCG by *istd* . For the definition of *istd* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).
    If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the range of possible tributary structures depends on interface standard specified for this VCG by *istd* and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). The signal will be immediately terminated on the unit. So no unequipped signal is

visible to the outside world.

The parameter *unequiposig* is set automatically according to the input value of the *inputsig* parameter if *modifier* = ES64U. For *modifier* = ES64U the parameter *unequiposig* is only retrievable and cannot be changed, because it follows always the value of the *inputsig* parameter.

Type: numeric string of the form ***NX***

- ***N*** specifies the count of container type ***X***
- ***X*** specifies the container type (or envelope).

Range:

Values	Modifier	Interface standard	Virtual concatenation mode	Represented signal structure
N = 7, ***X*** = 4	omitted	SDH	not applicable	7 × VC4
N = 7, ***X*** = 4	1GEPL	SDH	ENABLE	7 × VC4
N = 64, ***X*** = 4	10GEPL	SDH	ENABLE	64 × VC4
N = 1, ***X*** = c64	10GEPL	SDH	DISABLE	1 × VC4-64C
N = 21, ***X*** = 1	omitted	SONET	not applicable	21 × STS-1
N = 21, ***X*** = 1	1GEPL	SONET	ENABLE	21 × STS-1
N = 7, ***X*** = 3	1GEPL	SONET	ENABLE	7 × STS-3C
N = 192, ***X*** = 1	10GEPL	SONET	ENABLE	192 × STS-1
N = 64, ***X*** = 3	10GEPL	SONET	ENABLE	64 × STS-3C
N = 1, ***X*** = 192	10GEPL	SONET	DISABLE	1 × STS-192C

Range: For *modifier* = ES64U the same value set is applicable as for parameter *inputsig*.

Default:

Modifier	Interface Standard	Default Value
omitted	SDH	74
1GEPL	SDH	74
10GEPL	SDH	644
omitted	SONET	211
1GEPL	SONET	211
10GEPL	SONET	1921

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, that means, VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception:

For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.

26. **`vcat`** – Virtual Concatenation
    This parameter vcat specifies whether the Ethernet signal is mapped to a number of virtually concatenated tributaries (value = ENABLE) or into a contiguously concatenated tributaries (value = DISABLE). In the latter case, LCAS related parameters become insignificant (values set to default, values cannot be changed). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *vcat* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - For *modifier* ES64U: ENABLE (default)
    - otherwise: DISABLE.
    - For *modifier* = ES64U the parameter *vcat* is visible for retrieval, but cannot be modified once the VCG is established.
27. **`vmode`** – VCG Monitoring Mode
    the parameter *vmode* specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range:
    - MON, for monitored
    - NMON (default), for not monitored.
28. **`vpm`** – VCG Performance Monitoring
    The parameter *vpm* specifies whether the performance monitoring is enabled or disabled for this port.
    This parameter is not supported if the input parameter: *modifier* = ES64U. If the parameter *vpm* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - DISABLE (default)
    - ENABLE.
29. **`vrtswid`** – Virtual Switch Identifier
    This parameter identifies the instance of the virtual switch.
    This parameter is not supported if the input parameter *modifier* = ES64U. If the parameter *vrtswid* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 – Currently this is the only allowed value

30. **vvidlsd** – Valid VLAN Identifier List (dynamic)

    This parameter gives the list of all dynamic VLAN identifiers allowed for this port as gathered autonomously via GVRP. The list is ampersand grouped (& and &&). The vlan identifier of an incoming frame must be a member of this dynamic VLAN Id list or the static VLAN Id list vvidls. Frames tagged by a vlan identifier not member of these lists will be discarded. This parameter is also used in transparent tagging mode to report the list of valid customer tags (CIDs) for network ports (*cnprole* = NETWORK) collected by GVRP.

    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.

    If the parameter *vvidlsd* is generally not supported, then the parameter is always omitted. Otherwise, this parameter is reported only if

    - the virtual switch entity exists (i.e. the VLAN tagging mode vlantagmd of the slot the addressed EPort belongs to is set to 802.1TAG) AND
    - GVRP is enabled on the virtual switch (*gvrpmode* = ENABLED)
    - the port is a NETWORK port (*cnprole* = NETWORK).

    Type: Alphanumeric string and the character ’&’.

    Range:

    - Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
    - In case the list has no members (empty list) -1.

    If the value can not be retrieved: UNAVAILABLE

    Default: There is no default value.

    Example 1:

    vvidlsd=-1 The list has no members (empty list).

    Example 2:

    vvidlsd=3&5&40&&45 This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.

31. **wtr** – Wait To Restore Time

    The parameter *wtr* specifies the wait to restore time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it recovers from a faulty state then this tributary is taken in again to transmission by LCAS after the wait-to-restore time. This time is configured on group level, i.e. all tributaries have the same value for the wait-to-restore time. This parameter only affects NE behavior if *lcasmd* = ENABLED.

    For *modifier* = ES64U the parameter is visible with default values for retrieval, but cannot be modified.

    Type: Alphanumeric value set.

    Range: 0, 1, 2, …, 59, 60 [minutes].

    Default: 0.

Example output

Example of VCG creation related to a 1GE interface of a GE1 pack:

```
   UNITE-NE 08-03-28 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=08-03-28:ENT-VCG:1-1-#-#-2-
      V3::cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT,epor
;
```

Example of VCG creation related to a 1 GE interface of the GE10PL1 pack (interface standard mode SONET):

```
   UNITE-NE 08-03-28 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=08-03-28:ENT-VCG-1GEPL:1-1-#-#-39-v1::\
    enetpn=DEFAULT,enettca=DEFAULT0,eportClass=TRADITIONAL,\
    eportControl=NORMAL,htmr=0,inputsig=211,istd=SONET,lcasmd=ENABLE,\
    plcrthr=21,plctthr=21,tcamd=TR,unequiposig=211,vcat=ENABLE,\
    vmode=NMON,vpm=DISABLE,wtr=0,networkTNAType=IPV4,\
    networkTNAAddress=0.0.0.0,localLPID=0,clientNodeID=0.0.0.0,\
    networkTNATypeVCAT=IPV4,networkTNAAddrVCAT=0.0.0.0,localLPIDVCAT=0″
;
```

Example of VCG creation related to a 10 GE interface of the GE10PL1 pack (interface standard mode SDH):

```
   LT-UNITE 08-03-28 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=08-03-28:ENT-VCG-10GEPL:1-1-#-#-39-v9::\
    enetpn=DEFAULT,enettca=DEFAULT0,eportClass=TRADITIONAL,\
    eportControl=NORMAL,htmr=0,inputsig=644,istd=SDH,lcasmd=ENABLE,\
    plcrthr=64,plctthr=64,tcamd=TRRTR,unequiposig=644,vcat=ENABLE,\
    vmode=NMON,vpm=DISABLE,wtr=0,networkTNAType=IPV4,\
    networkTNAAddress=0.0.0.0,localLPID=0,clientNodeID=0.0.0.0,\
    networkTNATypeVCAT=IPV4,networkTNAAddrVCAT=0.0.0.0,localLPIDVCAT=0″
;
```

Example of VCG creation related to a STS-1 interface of the ES64U pack (interface standard mode SONET)

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ENT-VCG-ES64U:1-1-#-#-22-v9::\
    enetpn=DEFAULT,eportClass=TRADITIONAL,eportControl=NORMAL,\
    htmr=0,inputsig=11,istd=SONET,lcasmd=DISABLE, plcrthr=1,\
    plctthr=1,unequiposig=11,vcat=ENABLE,vmode=NMON,wtr=0,\
    clientNodeID=0.0.0.0,networkTNAType=IPV4,networkTNAAddress=0.0.0.0,\
    localLPID=0,networkTNATypeVCAT=IPV4,networkTNAAddrVCAT=0.0.0.0,\
    localLPIDVCAT=0″
```

;

ENT-VCG-TRANS notification

Output format

The **ENT-VCG-TRANS** notification uses the general format of a REPT DBCHG notification.

Syntax

ENT-VCG*:aid::spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values **vall** for Simple_VCGAID and **usall** for Simple_UnivSlotAID is accepted.

Spec block parameters

1. **camd** – Consequent Action Mode
   This parameter specifies the mode of the consequent action to be executed in egress direction (to the synchronous port) on detected signal fail conditions (dPLM, AI_TSF or detected generic AIS) on the ODU layer
   Type: Alphanumeric value set.
   Range: GENERICAIS (default), MSAIS, LASEROFF
2. **enetpn** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to a tributary ASAP profile name of type ENET for the designated AID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
3. **fevpm** – Far End VCG Performance Monitoring
   The parameter fevpm specifies whether the far end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore far end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
4. **inputsig** – Tributary Input Signal Rate List
   This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure is independent from the interface standard istd specified for this VCG.

Type: numeric string of the form NX

- N specifies the count of container type X
- X specifies the container type (or envelope).

Range:

- N=17,
- X = 4 for VC-4 / STS3-3c.

Default: 174.

5. **nevpm** – Near End VCG Performance Monitoring
The parameter nevpm specifies whether the near end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore near end PM on the ODUkP layer.
Type: Alphanumeric value set
Range: DISABLE (default), ENABLE
6. **istd** – Interface Standard
This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
Type: Alphanumeric value set.
Range:
   - SDH
   - SONET.

   Default: As derived from parameter *istddflt* specified at configuration management, section System RTRV-EQPT.
7. **nevpm** – Near End VCG Performance Monitoring
This parameter specifies whether the near end performance monitoring is enabled or disabled for this port.
Type: Alphanumeric value set.
Range: DISABLE (default), ENABLE
8. **pmvcgfeses** – Performance Monitoring VCG Far-End SES Threshold
This parameter describes the ODUkP Far-End SES threshold value. It indicates the number of errored blocks per second to declare a second as severely errored. The default value is defined in G.8201 for OTN. If the interface standard is changed the current default value is not changed.
Type: numeric value set.
Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
Default: ODU1P: 3064
G.8201 defines block based PM on the ODUkP layer.
9. **pmvcgneses** – Performance Monitoring VCG Near-End SES Threshold
This parameter describes the VCG (actually ODUkP) Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface

standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
Default: ODU1P: 3064
G.8201 defines block based PM on the ODUkP layer.

10. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable: TR-only, (also known as the implicit reset method). TR-RTR, (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: TRRTR in SDH mode; TR in SONET mode.

11. **`transpn`** – Transparent Alarm Severity Assignment Profile Name
This parameter points to a ASAP profile name of type TRANSPARENCY for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: 1 … 24 characters length
Default: DEFAULT.

12. **`transtca`** – Transparent TCA Profile Name
The parameter transtca specifies the name of the threshold crossing alert (TCA) profile used for the VCG..
Type: Alphanumeric string.
Range: 1 … 24 characters length
Default: DEFAULT0

13. **`unequiposig`** – Unequipped Tributary Output Signal Rate List
This parameter specifies the tributary structure and the capacity of the VCG port in output direction. As long as there is no cross-connection applied the specified tributaries are occupied by an unequipped signal. The parameter will implicitly give the maximum capacity of the VCG port, too. The tributary structure is independentof the interface standard specified for this VCG.
Type: numeric string of the form NX.
    - N specifies the count of container type X
    - X specifies the container type (or envelope)

    Range:
    - N = 17
    - X = 4 for VC-4 / STS-3C

    Default: 174

14. **`vmode`** – VCG Monitoring Mode
This parameter wtr specifies the monitoring mode for VCG.
Type: Alphanumeric value set.
Range: MON: for monitored; NMON (default): for not monitored.

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ”TIME=11-10-30,DATE=00-11-11:ENT-VCG-TRANS:1-1-#-#-2-V3::
      camd=GENERICAIS,enetpn=DEFAULT,fevpm=DISABLE,inputsig=174,nevpm=DISABLE,pmvcgfeses=3064,p
;
```

ENT-VCGTRIB notification

Output format

The **ENT-VCGTRIB** notification uses the general format of a **REPT DBCHG** Report Database Change notification.

Syntax: The **ENT-VCGTRIB** notification has the following specific syntax:

ENT-VCGTRIB-modifier:*aid*::*spec_block*;

Output parameters

Command parameters

1. **modifier**
   This parameter specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: STS, VC, STST, VCT.
2. **aid** – Access Identifier
   This parameter specifies the VCG tributary whose attributes are reported.
   Type: AID.
   Range: VCGNumAID as defined in AID structure with exception of value all.

Spec block parameters

1. **brstintvl**- Bursty Interval
   This parameter specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstintvl* has no affect to NE behavior.
   Type: Integer..
   Range: 2, 3, …, 10 [seconds].
   Default: 7 [seconds].
2. **brstthr** – Bursty Error Threshold
   This parameter specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30 [%].
3. **crslpbkstat** – Cross-connect Loopback Status
   This parameter reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).
   Type: Alphanumeric value set.
   Range: YES, for enabled loopback; NO, for disabled loopback.

4. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if sdsfmode is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   Type: Integer.
   Range: -5, -6, -7, -8, -9 i.e. range is 10exp-9 to 10exp-5.
   Default: -6
5. **`dexcthr`** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section.
   Type: Integer.
   Range: -3, -4, -5 i.e. range is 10exp-5 to 10exp-3.
   Default: -3.
6. **`exphptrc`** – Expected HP trace identifier
   This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval but cannot be modified.
   Type: Hexadecimal number without a preceding "H".
   Range: length, 32 hexadecimal digits; value, specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000.
7. **`feppm`** – Far End Path PM Enable
   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
   For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval but cannot be modified.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE.
   Default: DISABLE.
8. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion
   This parameter specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.

Range: DISABLE , ENABLE.
Default: The default value depends on interface standard specified by istd for the VCG owning this tributary. For definition of *istd* refer to "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078) Output parameters. DISABLE, if *istd* = SONET; ENABLE, if *istd* = SDH.

9. **`hptimmon`** – HP Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE.
10. **`inmbs`** – Input Membership
    This parameter specifies if the tributary shall take part to the transmission of its owning VCG in input direction. Input direction in this case means from the LCAS source on the GE1 pack to the cross-connect pack XC***n***. The addressing scheme of the tributaries by using VCGNumAIDs is ensuring that only legal AIDs are used. So no consideration about the "illegal" tributaries (for example, 1-1-#-#-1-1-3 for VC-4) is necessary. The parameter *inmbs* is only valid for modifiers different from STST and VCT, that means, is omitted in the command response for STST and VCT.
    The system shall not evaluate the parameter *inmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
    Type: Alphanumeric value set.
    Range: DISABLE , ENABLE.
    Default:
    - for modifiers STST or VCT: ENABLE
    - for all other modifiers: DISABLE.

    The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.
11. **`inmbst`** – Input Member Status
    This parameter reports the status of the tributary related to its VCG membership in input direction.
    Type: Alphanumeric value set.
    Range:
    - for modifiers STST or VCT always ACTIVE is reported
    - for all other modifiers:
    - NA, for not applicable, i.e. if the trib. is no member of the VCG or if virtual concatenation is disabled
    - ADD, the tributary is in the LCAS state ″Wait-for-Add″, ″Add″ or ″Wait-for-ACK″

- ACTIVE, the tributary is member of the VCG and working
- FAILED, the tributary is member of the VCG but taken off by LCAS (incoming failed condition for the sink tributary)
- UNAVAILABLE, for not available, i.e. the trib. is member of the VCG but the current data is not retrievable for any reason.

  The state FAILED can only be reported if *lcasmd* is set to ENABLE for the VCG owning this tributary. For parameter *lcasmd* refer to the Output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078).

  For VCG tributaries located on ES64U packs, only "NA" will be reported.

12. **neppm** – Near End Path PM Enable

    This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.

    For VCG tributaries located on ES64U packs, this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.

    Type: Alphanumeric value set.

    Range:

    - DISABLE (default)
    - ENABLE.

13. **outmbs** – Output Membership

    This parameter specifies if the tributary shall take part to the transmission of its owning VCG in output direction. Output direction in this case means from the cross-connect pack XCn to the LCAS sink on the GE1 GE***n*** pack. The parameter outmbs is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT. The system shall not evaluate the parameter *outmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.

    Type: Alphanumeric value set.

    Range:

    - DISABLE
    - ENABLE.

    Default: for modifiers STST or VCT ENABLE. for all other modifiers: DISABLE. The GE***n*** and ES64U has as default the size 0, i.e. it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user may decide explicitly if one tributary shall take part to the transmission.

14. **outmbst** – Output Member Status

    This parameter reports the status of the tributary related to its VCG membership in output direction.

    Type: Alphanumeric value set.

Range:

- NA, for not applicable, if i.e. the trib. is no member of the VCG or if virtual concatenation is disabled
- ADD, the tributary is in the LCAS state ″Wait-for-Add″, ″Add″ or ″Wait-for-ACK″
- ACTIVE, the tributary is member of the VCG and working
- FAILED, the tributary is member of the VCG but not working due to a failure condition. Includes period during ″Hold-Off″.
- WTR, failure condition has cleared, tributary is in wait to restore condition.
- UNAVAILABLE, for not available, i.e. the trib. is member of the VCG but the current data is not retrievable for any reason.

  The state WTR can only be reported if lcasmd is set to ENABLE for the VCG owning this tributary. For parameter lcasmd refer to RTRV-VCG Output Parameters. For a detailed explanation of the ADD state in the LCAS protection scheme, please refer to SRD_DATA_Virtual_Concatenation.

For VCG tributaries located on ES64U packs, only “NA” will be reported.

15. **`pmhpfeses`** – PM HP Far End SES Threshold

    This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). If *istd* is not set to SDH then *pmhpfeses* has no affect to NE behavior.

    For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.

    Type: Integer.

    Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.

    Default: 2400.

16. **`pmhpneses`** – PM HP Near End SES Threshold

    This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). If *istd* is not set to SDH then *pmhpneses* has no affect to NE behavior.

    For VCG tributaries located on ES64U packs, this parameter can be modified

although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
Default: 2400.

17. **`ptca`** – Path TCA Profile Pointer
This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.
The parameter *ptca* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
Type: Alphanumeric string.
Range: 1 … 24 characters.
Default: DEFAULT0.

18. **`ptpn`** – Path Terminating ASA Profile Name
This parameter points to a tributary ASAP profile name pfname for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: 1 … 24 characters.
Default: DEFAULT.

19. **`ptrcrfmt`** – Path Trace Read Format
This parameter specifies the format of the path trace (J1 byte) to be received (sink).
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: numeric string.
Range:
- 2, for non-specific repeating byte
- 16, for 16 byte specific string
- 64, for 64 byte string.

Default:
For VCG tributaries located on GE1 packs: 2
For VCG tributaries located on GE10PL1 or ES64U packs: 16 if *istd* = SDH, 64 if *istd* = SONET.

20. **`ptrcwfmt`** – Path Trace Write Format
This parameter specifies the format of the path trace (J1 byte) to be sent (source).
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: numeric string.

Range:

- 2, for non-specific repeating byte
- 16, for 16 byte specific string
- 64, for 64 byte string.

Default:
For VCG tributaries located on GE1 packs: 2
For VCG tributaries located on GE10PL1 or ES64U packs: 16 if *istd* = SDH, 64 if *istd* = SONET.

21. **sdsfmode** – Signal Degrade/Signal Fail Mode
This parameter specifies the determination mode of degraded defect for transmission.
Type: Alphanumeric value set.
Range: BURST or POISSON.
Default: For VCG tributaries located on a GE10PL1 pack, the default is: BURST.
For VCG tributaries located on other packs, the default value depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).
    - POISSON, if *istd* = SONET
    - BURST, if *istd* = SDH
22. **tcamd** – TCA mode
This parameter is only available for SDH mode. This means the modifier must be VC. The parameter shows the modus of the PM thresholding. Two thresholding methods are applicable for SDH: TR-only (also known as the implicit reset method), TR-RTR (also known as the explicit reset method). For SONET the parameter *tcamd* does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: SDH mode: TRRTR. SONET mode: TR (internal).
For SONET the parameter *tcamd* does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
The parameter *tcamd* is only valid for modifiers different from STST and VCT, i.e. shall be omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.
23. **tmonmd** – Tributary Monitoring Mode
This parameter specifies the monitoring mode for the tributary.
For VCG tributaries located on ES64U packs, this parameter can be modified although it has currently no effect.
Type: Alphanumeric value set.
Range: MON or NMON (default).

24. **`trc`** – Outgoing Path Trace
This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
Type: Hexadecimal number without a preceding "H".
Range: the range depends on the parameter ptrcwfmt specified in **`RTRV-VCGTRIB`** Output parameters length
    - 32, hexadecimal digits, if *ptrcwfmt* = 16
    - 128, hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt* = 16; 64 byte string, if *ptrcwfmt* = 16.

    Default: Depending on the default value of *ptrcwfmt*:
    - 00hex, if *ptrcwfmt* = 2.
    - 89hex followed by 15 bytes 00hex, if *ptrcwfmt* = 16.
    - 64 bytes 00hex, if *ptrcwfmt* = 64.
25. **`tslmmon`** – Trail Signal Label Mismatch Monitor
This parameter specifies the mode of the TSL mismatch detection.
For VCG tributaries located on ES64U packs, this parameter is visible with default values for retrieval, but cannot be modified.
Type: Alphanumeric value set.
Range:
    - DISABLE
    - ENABLE (default).

Example output

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ENT-VCGTRIB-STS:1-1-#-#-2-V1-4::
      brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-
      3,hptimais=DISABLE,exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,inmbs=DISAB
;
```

Creation of VCGTRIB entities for a 10 GE interface on a GE10PL1 (SDH interface standard):

```
   LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:ENT-VCGTRIB-VC:1-1-#-#-1-v9-1&&-192::\
    brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=3,\
```

```
    hptimais=DISABLE,exphptrc=8900000000000000000000000000000,\
    hptimmon=DISABLE,inmbs=DISABLE,inmbst=NAV,outmbs=DISABLE,\
    outmbst=NAV,ptpn=DEFAULT,ptrcrfmt=16,ptrcwfmt=16,sdsfmode=BURST,\
    tmonmd=NMON,trc=00,tslmmon=ENABLE”
;
```

ENT-VRTSW notification

Output format

The **ENT-VRTSW** notification uses the general format of a REPT DBCHG (Report Database Change) notification.

Command syntax: **ENT-VRTSW***:aid:vrtswid:spec_block;*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the slot containing a GE1 pack to be operated on.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure with the exception of the value all and usall.
2. **vrtswid** – Virtual Switch Identifier
   This parameter identifiers the instance of the virtual switch.
   - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
   - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

   Type: Integer.
   Range: 1 – Currently this is the only allowed value.
   Default: 1.

Spec block parameters

1. **eplst** – EPORT AID list
   This parameter specifies a list of EPORT instances connected to the virtual switch instance as specified by *vrtswid*.
   In R3 there is only one virtual switch instance to which all ports are mapped. Therefore this list is fix.
   Type: AID list.

Range: A list of EPortAIDs as specified in the AID definition section with the exception of the values all and usall in the respective components of the AID, separated by an ampersand ('&')
Example: If all four EPORTs of a GE1 are mapped to one virtual switch: 1-#-#-1-1-1&1-#-#-1-1-2&1-#-#-1-1-3&1-#-#-1-1-4

2. **`ethertype`** – VLAN tag protocol ID
This parameter specifies the VLAN tag protocol ID necessary for interworking purposes. This parameter is used to control whether the virtual switch operates in IEEE802.1Q tagging mode (i.e. set to 0x8100) or in transparent tagging mode (i.e. set to value different to 0x8100).
Type: Hexadecimal value.
Range: 0x0600 .. 0xFFFF.
The following values have a specific meaning:
   - 0x8100: virtual switch operates in IEEE802.1Q tagging mode
   - any other value: virtual switch operates in transparent tagging mode.

   Default: 0x8100.
3. **`gvrpmode`** – GVRP mode
The parameter *gvrpmode* allows to control the dynamic VLAN registration mode.
Type: Alphanumeric value set.
Range: ENABLED, GVRP is enabled. DISABLED (default), GVRP is disabled.
4. **`stbrpriority`** – Spanning tree bridge priority
This parameter specifies the priority of the bridge to be used by the Spanning Tree Protocol.
Type: Integer.
Range: 0 to 65535.
The stepping as recommended by the standard (steps of 4096 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 61440″ is not done by the NE.
Default: 32768.
5. **`stfdtm`** – Spanning Tree forward delay timer
The parameter *stfdtm* specifies the rSTP-related bridge forward delay timer. This parameter is relevant for both rSTP and STP.
Type: Integer.
Range: 1 .. 255.
Default: 15.
6. **`sthellotm`** – Spanning Tree hello timer
The parameter *sthellotm* specifies the rSTP-related bridge hello timer. This parameter is relevant for both rSTP and STP.
Type: Integer.
Range: 1 .. 225.
Default: 2.

7. **`stmaxage`** – Spanning Tree maximum age timer
   The parameter *stmaxage* specifies the rSTP-related maximum PDU transmission delay timer. This parameter is relevant for both rSTP and STP.
   Type: Integer.
   Range: 1 .. 255.
   Default: 20.
8. **`vcglst`** – VCG AID list
   The parameter *vcglst* specifies a list of VCG instances connected to the virtual switch instance as specified by *vrtswid*.
   In R3 there is only one virtual switch instance to which all eports are mapped. Therefore this list is fix.
   Type: AID list.
   Range: A list of VCGAIDs as specified in the AID definition section with the exception of the values all, usall and vall in the respective components of the AID, separated by an ampersand ('&').
   Example: If all four VCGs of a GE1 are mapped to one virtual switch:
   1-#-#-1-1-v1&1-#-#-1-1-v2&1-#-#-1-1-v3&1-#-#-1-1-v4.

Example output

```
  LT-UNITE 02-05-15 13:31:30
A  001 REPT DBCHG
   "TIME=13-30-00,DATE=02-05-15:ENT-VRTSW:1-1-#-#-1:1:eplst=1-1-#-#-1-1&1-1-#-#-1-
      2&1-1-#-#-1-3&1-1-#-#-1-
      4,gvrpmode=ENABLED,stbrpriority=32768,stfdtm=15,sthellotm=2,stmaxage=20,ethertype=0x8100,
;
```

INH-MSG: Inhibit message

The **INH-MSG** command instructs the NE to inhibit the forwarding of autonomous messages on the user session or per system (depending on parameter *sessions*). Autonomous messages are messages which are provided by the system itself without a trigger via the management interface (e.g. alarms, switch reports).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: no

Input format

Command syntax: **INH-MSG-ALL**:*tid*::*ctag*[::,,,,*sessions*];

The purpose of the commas in case the optional parameter *sessions* is present is that the format is compliant with non-supported position defined parameters in GR-833.

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **sessions** – Affected sessions
   This parameter specifies for which sessions the *msgstate* is applied.
   Type: Alphanumeric value set.
   Range: CURRENT: affected session is the current session of the user who executed the command.
   Default: CURRENT

Output format

If the INH-MSG request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command disables autonomous message reporting as specified by *sessions*.

If the user is logged in more than once, and *sessions* = CURRENT, only the current login session is affected by this command.

If the **INH-MSG** command inhibits autonomous messages that have already been suspended, it will complete successfully.

For *sessions* = CURRENT, the *atag* sequence will be reset to 000 when autonomous message are resumed.

The *atag* for a memory administration user is not affected by this command, see also "RTRV-AO: Retrieve autonomous output" (p. 3-2241).

Example input/output

The following is an example of the **INH-MSG** command:

```
INH-MSG-ALL:LT-UNITE::123456;

   LT-UNITE 00-11-15 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **INH-MSG** command.

Related TL1 messages

- "ALW-MSG: Allow message" (p. 3-33)
- "RTRV-ATTR-MSG: Retrieve attribute message" (p. 3-2260)

References

GR-833-CORE

INIT-EQPT (slot): Initialize equipment (slot)

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No.

Input format

Command syntax: **`INIT-EQPT`**:*tid*:*aid*:*ctag*[:::*spec_block*];

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   The AID of a slot.
   Type: AID.
   Range:
   - SlotAID
   - UnivSlotAID
   - XCSlotAID
   - CTLSlotAID.

   The keyword **`usall`** is not allowed for Simple_UnivSlotAID
   The keyword **`xcall`** is not allowed for Simple_XCSlotAID
   The keyword **`ctlall`** is not allowed for Simple_CTLSlotAID
   The keyword **`all`** is not allowed for Simple_SlotAID
   The keyword **`all`** is not allowed on bay/shelf level.
3. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`resettype`** – Reset Type
   With this parameter the type of the reset is selected. Full reset can be traffic affecting and controller reset is not traffic affecting.
   Type: Alphanumeric value set.

Range:

- FULL: Full reset
- CONTR: Controller reset.

Default: FULL.

Output format

If the NE fully complies with the INIT-EQPT request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

- *resettype* = FULL: The command resets the controller and all transmission devices of the circuit pack in the addressed slot.
- *resettype* = CONTR: The command resets only the controller of the circuit pack in the addressed slot and not the transmission devices.

The **INIT-EQPT** command does not generate a **REPT DBCHG** message. There are no unique state change notifications associated with the **INIT-EQPT** command. There are, however, all the state change notifications (operational state) associated with the subsequent restart of the circuit pack.

Update affected entities

These resets affect the operational state of the circuit pack. This change is reported by means of an ED-STATE-EQPT notification for the affected circuit pack. An exception is the reset for CTL pack. For this board no notification is created.

Example input/output

The following shows an example of the **INIT-EQPT** command where the AID identifies the universal slot 7 in bay 1, shelf 1:

```
INIT-EQPT:LT-UNITE:1-1-#-#-7:123456:::resettype=CONTR;

   LT-UNITE 00-11-16 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **INIT-EQPT** command.

Related TL1 messages

"INIT-SYS: Initialize system" (p. 3-2026)

References

GR-833-CORE

INIT-SYS: Initialize system

The **INIT-SYS** command results in a full reset of all circuit packs that are present in the system. The initialization is equivalent to power cycle reset of the whole system.

The **INIT-SYS** command is traffic affecting.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: no

Input format

Command syntax: **INIT-SYS**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the NE fully complies with the INIT-SYS request, the following normal completion response is returned to indicate the initialization has been started:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **INIT-SYS** command does not generate a **REPT DBCHG** message. There are no unique state change notifications associated with the **INIT-SYS** command. There are, however, all the state change notifications associated with the subsequent restart of all CTIPs. This includes multiple state change messages. Refer to autonomous notifications because of state changes.

Example input/output

The following is an example for the **INIT-SYS** command:

```
INIT-SYS:LT-UNITE::123456;

   LT-UNITE 00-11-16 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to INIT-SYS command.

Related TL1 messages

“INIT-EQPT (slot): Initialize equipment (slot)” (p. 3-2023)

References

GR-833-CORE

OPR-DEFRAG: Perform defragmentation

Overview

This command is used to initiate the defragmentation of a link or other entity.

This command is issued by an operator to actually perform a defrag or empty port operation on a port. The command causes the receiving node to start the defrag or empty port operation as specified by the scope field. The sequence of operations that will be followed is first computed and then each action is carried out as an atomic operation.

Database change reports generated

As a result of the time-slot movements, this command shall cause db change notifications.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P3

Abortable: No.

Input format

Command syntax: **`OPR-DEFRAG:`***src-Tid:[src-Aid]:ctag::scope:targetID*;

Input parameters

1. **`srcTid`** – Target Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

3. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
4. **`scope`**
   This parameter specifies the scope of the defrag or empty port operation: intra-port,inter-port [intra link] or empty port.
   Type: Alphanumeric value set.

   Range:

   - PORT: specifies that the defrag is intra fiber, i.e. on a specific port. In this case the targetID is the same as the portAID. The corresponding neighbor node and neighbor portAID can be deduced form the targetID.
   - LINK: specifies the link defrag is inter fiber but within a link bundle. Also, the link defrag operation is performed on all the bundles between the two neighboring nodes. TargetID specifies the neighbor nodes ASTN node ID.
   - EMPTYPORT: specifies to empty a port by moving all connections via this port to other ports within the same link bundle. The targetID defines the port which should be emptied. The corresponding neighbor node and neighbor portAID can be deduced form the targetID.
5. **`targetID`**
   The specification of the targetID depends on the setting of the parameter scope: PORT specifies the AID of the port on the local node that is to be defragmented LINK specifies the ASTN node ID of the neighbor node of the connected link that is to be defragmented EMPTYPORT specifies the AID of the port on the local node that is to be emptied
   Type: Four dash separated numeric string for the portAID. Four dot separated numeric string for the neighbor ASTN node ID.
   Range: Each numeric string: 0…255.

Output format

If the **`OPR-DEFRAG`** request checked on the input parameters against NNDM, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
;
```

The final successful defragmentation operation and also the interruption or abortion of the operation is sent out via **`REPT NNCHG`** notification

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

Example input/output

Input/output examples for intra-port defragmentation

The following example shows a successful completion of a intra-port defragmentation command:

```
OPR-DEFRAG:NN001:snn:ed01::PORT,1-1-2-1;

   NN001 2002-01-12 01:03:16
M  ed01 COMPLD
;
```

Input/output examples for inter-port defragmentation

The following example shows a successful completion of an inter-port defragmentation:

```
OPR-DEFRAG:NN001:snn:ed01::LINK, 29.13.80.192;

   NN001 2002-01-12 01:03:16
M  ed01 COMPLD
;
```

The following example shows a successful completion of an empty port defragmentation:

```
OPR-DEFRAG:NN001:snn:ed01::EMPTYPORT,1-1-2-1;

   NN001 2002-01-12 01:03:16
M  ed01 COMPLD
;
```

Error responses

The following list identifies conditions that will cause a similar error response:

`E410 (IIPI) Input, Invalid PortId`

`E512 (IINI) Error – Input, Invalid Neighbor node ID`

Conflicting Defragmentation Request

There is already a defragmentation action in progress with the NE. The following list identifies conditions that will cause a similar error response:

Error code: `E620 (DAIP) Defragmentation already in progress`

Port Failure

If a failure occurs like remote node crash or fiber cut while a defrag or empty port operation is requested, the request will return with a DENY message and an error code specifying the type of failure. The following list identifies conditions that will cause a similar error response:

Error code: `E617 (PLFS) Port or Line in Failed State`

Port Lockout State

Depending on the operation request the system will behave differently if a port is in LOCKOUT state:

- Defrag Operation: The request will return with a DENY message and an error code specifying the type of failure if one of the ports is in LOCKOUT state.
- Empty Port Operation: The request will return with a DENY message and an error code specifying the type of failure if the specified port is not in LOCKOUT state. Other ports of the link bundle are allowed to be in LOCKOUT state.

The following list identifies conditions that will cause a similar error response:

Error code: `E618 (PLOS) Port in Lock-Out State`

Error code: `E619 (PNLS) Port is not in Lock-Out State`

Port can not be emptied

The request will return for an empty port operation with a DENY message and an error code if:

- it is the only port in the link bundle
- all other ports of the link bundle are in LOCKOUT state
- no connections can be moved, because there is no place in the other (not locked-out) ports of the same link bundle.

The following list identifies conditions that will cause a similar error response:

- `E621 (NTPA) No Target Port Available`
- E622 (NFSA) No Free Space on remaining link bundle Available

Related TL1 message

- "RTRV-PREDEFRAG: Perform an ONNS fragmentation analysis" (p. 3-2738)

OPR-EXT-CONT: Operate external control

The **OPR-EXT-CONT** command operates a MDO port (Miscellaneous Discrete Output / external discrete control), that means, it switches on the MDO port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **OPR-EXT-CONT**:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDO port entities on which the command operates. If misc_outall is specified the configuration of all MDO's will be retrieved.
   Type: AID
   Range: Miscellaneous_out_discreteAID
   Default: misc_outall (if the *aid* is omitted).
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

The following example shows the successful completion of the **OPR-EXT-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

If the **OPR-EXT-CONT** command does not alter the existing attributes, i.e., the control referenced by the aid is already operated, the NE will not deny the command. The system will respond with the completion message shown above.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

If no probable cause is assigned to the MDO port the command changes the state of the MDO specified via aid to ″switched on″, i.e. the MDO port will be activated.

The update will be reported via a OPR-EXT-CONT – REPT DBCHG message. Refer to the definition of the OPR-EXT-CONT notification.

Example input/output

The following example shows the successful completion of the **OPR-EXT-CONT** command by the NE:

```
OPR-EXT-CONT:LT-UNITE-123:MISC_OUT3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **OPR-EXT-CONT** command.

Validation rule: probable cause assigned to MDO port

A probable cause is assigned to the MDO port. This command will be not performed.

Error code: `SDNC – Status, Data Not Consistent, asgncond not empty`

Related TL1 messages

- "RTRV-ATTR-CONT: Retrieve attribute control" (p. 3-2254)
- "SET-ATTR-CONT: Set attribute control" (p. 3-3133)
- "RLS-EXT-CONT: Release external control" (p. 3-2190).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

OPR-EXT-CONT notification

Output format

The **`OPR-EXT-CONT`** notification has the following specific syntax:

`OPR-EXT-CONT`:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Refer to the *aid* parameter in the output parameters section of the "RTRV-ATTR-CONT: Retrieve attribute control" (p. 3-2254).

Example output

The example shows a normal OPR-EXT-CONT notification:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   "TIME=08-00-12,DATE=00-01-01:OPR-EXT-CONT:MISC_OUT3"
;
```

OPR-LPBK: Operate cross-connection loopback

The **OPR-LPBK**-*modifier* command can be initiated by a user to establish a loopback on a high-order or low-order tributary.

In this section only the loopback on tributaries is considered.

A cross-connection loopback consists of a cross-connection from an input (high-order or low-order) tributary to an output (high-order or low-order) tributary of the same tributary via the referring cross-connection fabric. Note that this loopback is possible on any logical tributary whether there is a cross-connection for that tributary or there is no cross-connection for that tributary (in this case the tributary is said to be ″idle″ – thus the naming cross-connection loopback might be slightly misleading – the naming cross-connection loopback should be understood in the sense that the cross-connection fabric is involved in this command and not in the sense that an existing cross-connection is looped back) After releasing the loopback the former, pre-existing cross-connections (if any) must be automatically re-established.

For setting a loopback on an existing cross-connection the command parameter *cmdmde* has to set to FRCD, else if the tributary is *idle* the *cmdmde* is NORM.

The *modifier* in the **OPR-LPBK** command indicates the rate on which the loopback command is applied to. For high-order tributaries, there is no difference if the rate is specified as a SONET rate (STS1, STS3,..) or as SDH-rate (VC3, VC4,..) i.e. the **OPR-LPBK**-<*sonet-rate*> yields to the same physical and logical behavior as **OPR-LPBK**-<*sdh-rate*>. No check is made if the *modifier* matches the interface-standard of the port i.e. if the port is provisioned as SONET (resp. SDH) and the rate is a SONET (resp. SDH) rate.

For low-order tributaries this is not applicable as there are no tributaries of the same rate in SONET and SDH. Note that for high-order tributaries loopbacks can be still established (like ″idle″ tributaries) if they are substructured or even cross-connected on low-order level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No.

Input format

Command syntax: **OPR-LPBK**-*modifier*:*tid*:*aid*:*ctag*::,,,*lpbktyp*[:*spec_block*];

Input parameters

Command parameters

1. **`modifier`**
   This parameter indicates the tributary- respectively cross-connection-rate on which the **`OPR-LPBK`** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - LOVC3
   - LOVC12
   - LOVT1.

   No use of the **`all`** value is allowed.
2. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
3. **`aid`** – Access Identifier
   This parameter specifies the tributary for which the loopback is requested.
   Type: GenTribAID or GenLOTribAID.
   No **`all`** or **`usall`** or **`vall`**-value is possible for any part of the aid.
4. **`ctag`** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
5. **`lpbktyp`** – Loopback Type
   This parameter specifies the type of loopback to be operated. In case of a tributary loopback (either with or without cross-connection) this parameter will be set to CRS.
   Type: alphanumeric value string.
   Range: CRS.

Spec block parameters

1. **`cmdmde`** – Command Mode
   This parameter specifies the command execution mode. If the tributary is involved in a cross-connection the parameter is set to FRCD, else if the tributary is idle (i.e. non cross-connected) *cmdmde* is set to NORM.
   The difference is mainly that in case of an existing cross-connection (in_aid → out_aid) the counterpart (i.e. the out_aid) which formerly delivered traffic will be set to AIS by the loopback operation.
   Type: Alphanumeric value string.
   Range:
   - NORM (default)
   - FRCD.

Output format

If the **`OPR-LPBK`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Loopback on tributaries can be switched via the **`OPR-LPBK`** command. The parameters *crslpbkstat* and *crslpbkrate* of the entity tributary are updated: – *crslpbkstat* is updated to YES, – *crslpbkrate* is updated to the entered value as given by the *modifier* parameter.

The update of crslpbkrate is not applicable to LONIM entities as the rate is unique per *aidtype*.

This update is reported by an ED-*rr* (*rr* being either VC3 or STS1) notification for normal tributaries, an ED-VCGTRIB notification for virtual concatenation tributaries, an ED-LONIM notification for low order tributaries..

Behavior for ONNS

No cross-connection loopbacks are supported on ONNS ports (of port class, I-NNI, UNI, or E-NNI) as it is described in the validation rules below.

Cross-connection loopbacks on ONNS ports of port class EDGE are supported.

Updated affected entities when switching a HO-Loopback

Apart from the tributary entity no further entities are affected by the **OPR-LPBK** command.

When using the loopback possibility for VCG-tribs it is in the operators responsibility to ensure that the complete bundle of VCG-tribs (building the VCG) are looped. Otherwise parts of the traffic will be missing.

Update of affected entities when switching a LO-Loopback

In case of switching a LO-loopback while no related LO-Cross-connection is existing the related HO-tributary may be substructured automatically for the first time or the substructure may be adapted. For these side effects the same rules as for creating a LO-cross-connection is applied, see Detailed Behavior of ENT-CRS, ID 72712 and following.

Example input/output

Example 1:

Suppose the following aid 1-1-#-#-1-1-4 is *in_aid* of a cross-connection.

The following command will set a loopback for that tributary.

```
OPR-LPBK-VC4:UNITE-NE:1-1-#-#-1-14:123456::,,,CRS:
cmdmde=FRCD;

   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 2:

Suppose the following aid 1-1-#-#-1-1-4 is not cross-connected.

The following command will set a loopback for that tributary.

```
OPR-LPBK-VC4:UNITE-NE:1-1-#-#-1-1-4:123456::,,,CRS;

   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Valid rate

Loopback command processing verifies that the specified rate is equivalent to the rate of the existing cross-connection. Hereby the SONET-rates are treated identical to the SDH-rates (i.e. sts1/vc3, STS3/vc4 etc.).

The validation is only applied, if the specified tributary is consistent with the tributary boundaries for the specified rate.

Error code:

`IDNC – Input Data Not Consistent, Rate is inconsistent with existing cross-connection`

For the check of the tributary boundary consistency, separate validation rules are defined, see ID 29449 and ID 72755.

Validation rule: Valid loopback type

Loopback command processing verifies that the parameter and *lpbktyp* match i.e. for a tributary loopback the modifier must be present, the aid must specify a tributary and the *lpbktyp* must be CRS.

Error code:

`IDNC – Input Data Not Consistent, Inconsistent loopback type w.r.t. aid`

Validation rule: Loopback must not already exist

Loopback command processing verifies that the loopback command can not be applied to an already existing loopback.

Error code: `SAOP – Status Already Operated, loopback already exists`

Validation rule: No test access session

Loopback command processing shall deny commands requesting a loopback at a tributary which is used in a test access session either as tributary under test or as test access tributary.

Error code:

`SNVS – Status, Not in Valid State, Tributary used in test access session`

Validation rule: No HO loopback while LO cross-connection, substructure or conversion cross-connection exists

If the high-order tributaries specified in the high-order loopback command are substructured or LO cross-connections or conversion cross-connections exist at the tributaries, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections or substructure still exist`

Validation rule: No HO loopback while conversion cross-connection exists

If the high-order tributaries specified in the high-order loopback command are involved in conversion cross-connections, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`

Validation rule: No LO loopback while conversion cross-connections exist

If an attempt is made to create a LO loopback, while a conversion cross-connection exists at the HO tributary or at a LO tributary related to the same HO tributary, then the command is denied.

Error code:

`SNVS – Status, Not in Valid State, conversion cross-connections are still existing`

Validation rule: AID boundary and rate consistency

Loopback command processing verifies that the logical AID specified in the command is consistent with the modifier rate. Consistency implies the following condition:

- The boundary of the specified tributary is consistent with the specified rate.
- STS-1/VC-3 payloads can start at any position.
- The first (STS-1 equivalent) tributary of an STS-3/VC-4 rate cross-connection must be on a mod3 + 1 boundary (1, 4 , 7, …).
- The first STS-1 of an STS-12/VC-4-4c rate cross-connection must be on a mod12 + 1 boundary (1, 13, 25, …).
- The first STS-1 of an STS-48/VC-4-16c rate cross-connection must be on a mod48 + 1 boundary (1, 49, 97, …).
- The first STS-1 of an STS-192/VC-4-64c rate cross-connection must be on a mod192 + 1 boundary (1, 193, 385, …).

Error code:

`IDNC – Input Data Not Consistent, Tributary boundary is inconsistent with the indicated rate`

Validation rule: LO-AID boundary and rate consistency

Loopback command processing verifies that the logical AID specified in the command is consistent with the modifier rate. Consistency implies the following condition:

- The boundary of the specified tributary is consistent with the specified rate.
- STS-1/VC-3 payloads can start at any position.
- If modifier=LOVT1 then LONUMAID (1,2, … 28)
- If modifier==LOVC3 then LONUMAID (1, 22, 43).
- If modifier==LOVC12 then LONUMAID (1, 2, …).

Error code:

`IDNC – Input, Data Not Consistent, Tributary boundary is inconsistent with the indicated rate`

Validation rule: Entities must be adaptable for LO loopbacks

Loopback command processing shall verify that the low-order tributaries specified in the low-order loopback command can be properly created by adaptation of the existing substructure.

Error code:

`IENE – Input, Entity Not Exists, Substructure cannot be adapted`

Validation rule: Tributary provisioning and rate consistency

For fixed rate tributaries, loopback command processing verifies that

- If the port identified by logical input AID is provisioned to operate in the fixed signal rate mode, as determined by the tributary mode parameter, then the specified modifier rate is equivalent to the provisioned signal rate of the input tributary, as determined by the tributary input signal rate parameter.
- This also holds for the output tributary, which must match the input tributary in the sense of equivalence of modifier rate with the output signal rate Note that in both cases that SDH- and SONET tributary signal rates are treated identical (from the loopback perspective) i.e. there is no difference if the SDH values or the SONET values (for the inputsig- and unequiposig-parameters) are used for the tributary signal rate as long as those rates are compatible (i.e. vc3/sts1, vc4/sts3 etc.).

Error code:

`IDNC – Input Data Not Consistent, Specified Tributary does not support the indicated rate`

Note: In case of adaptive rate no consistency check is required

Validation rule: *cmdmde* matches cross-connection non-/existence

Loopback command processing will verify that

- in case no cross-connection exists the *cmdmde* must be NORM
- in case a cross-connection exists the *cmdmde* must be FRCD.

In the other cases

- no cross-connection and *cmdmde* = FRCD and
- cross-connection and *cmdmde* = NORM a rejection takes place.

Error code:

`IDNC – Input Data Not Consistent, cross-connection-existence not consistent with cmdmde-value`

Validation rule: No low-order crossconnection loopbacks with LOXC/1

Loopback command processing shall verify that the low-order switching pack currently used by the system is *not* of type LOXC/1.

Error code:

`SNVS – Status, Not in Valid State, low-order crossconnection loopbacks not allowed with LOXC/1`

Validation rule: No loopbacks on tributaries below ports of *portClass* INNI

Loopback command processing shall verify that the tributary does not belong to a port of *portClass* INNI.

Error code:

`SNVS – Status, Not in Valid State, Port must not be of portClass INNI`

Validation rule: No loopbacks on tributaries below ports of *portClass* UNI

Loopback command processing shall verify that the tributary does not belong to a port of *portClass* UNI.

Error code:

`SNVS – Status, Not in Valid State, Port must not be of portClass UNI`

Validation rule: No loopbacks on tributaries below ports of *portClass* ENNI

Loopback command processing shall verify that the tributary does not belong to a port of *portClass* ENNI.

Error code:

`SNVS – Status, Not in Valid State, Port must not be of portClass ENNI`

Validation rule: No low-order loopbacks with LOXC cap. equal 0

When creating lower-order loopbacks, the command processing shall verify that the capacity of the low-order switching pack is set to a value greater than 0.

Error code:

`SNVS – Status, Not in Valid State, LOXC switch capacity invalid`

Related TL1 messages

- “RLS-LPBK: Release loopback (facility)” (p. 3-2198)
- “RTRV-LPBK (facility): Retrieve loopback (facility)” (p. 3-2511)

References

GR-199-CORE

OPR-LPBK notification (cross-connection loopback)

There will be no **OPR-LPBK** notification because the changes (for the parameter *crslpbkstat* and *crslpbkrate*) caused by an OPR-LPBK command will be notified via an ED-*rr* (*rr* either VC3 or STS1) notification for normal tributaries, an ED-VCGTRIB notification for virtual concatenation tributaries, an ED-LONIM notification for low order tributaries.

OPR-LPBK: Operate facility loopback

For the GE***n***-pack only a far-side facility loopback for the EPort is allowed – an OPR-LPBK for the EPort to switch a near-side facility loopback is denied.

The **`OPR-LPBK`** command can be initiated by a user to establish a loopback on an OP***n*** or GE***n*** port (but not on an acgPort on LOXC packs) in the NE.

Two types of facility loopback can be distinguished:

- a near-side facility loopback which means that signal on the in-port is looped to the out-port and
- a far-side facility loopback, which means that the signal on the out-port is looped back to the in-port.

Switching a facility loopback has no restrictions to existing cross-connections loopbacks or cross-connections, i.e. facility loopbacks are assumed to precede over already existing cross-connection loopbacks. When a facility loopback is released the previously existing cross-connection loopback(s) will show up again.

Note that as safeguard for unintentional corruption of traffic the switching of a facility loopback will only be possible if the affected port's service condition is set to out of service (*servicecond* = OUTOFSERVICE) – this can be achived via the ED-*rr* command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **`OPR-LPBK`**-*modifier*:*tid*:*aid*:*ctag*::,,,*lpbktyp*;

Input parameters

Command parameters

1. **`modifier`**
   This parameter indicates the port rate on which the **`OPR-LPBK`** command acts on.
   Type: Alphanumeric value set.

   Range:

   - STM1E
   - STM1
   - STM4
   - STM16

- STM16T
- STM64
- STM256
- EC1
- OC3
- OC12
- OC48
- OC48T
- OC192
- OC768
- 1GE
- 1GEPL
- 10GEPL
- T3

No use of the **all** value is allowed.

2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   Access Identifier of the port for which the loopback state is returned.
   Type: AID
   Range: SPortAID (when using OP***n***, EP155, OPT***n*** packs or EC1 interfaces on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_SPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   APortAID (when using DS3 interfcaes on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_APortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   EPortAID (when using GE***n*** packs) as defined in AID structure. The values:
   - all for Simple_EPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.
4. **ctag** – Correlation Tag
   his is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

5. **`lpbktyp`** – Loopback Type
   This parameter specifies the type of loopback to be operated.
   In case of a facility loopback this parameter is set to
   - FACILITY for a near-side facility loopback
   - FSF for a far-side facility loopback.

   Type: Alphanumeric value string.
   Range:
   - FACILITY
   - FSF.

Output format

If the **`OPR-LPBK`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

Behavior – OP*n* Port

Loopback on OP***n*** ports can be switched via the **`OPR-LPBK`** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port are updated by this command:

- *lpbk* should be set to YES in case the *lpbktyp* = FACILITY
- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an ED-*rr* notification for an OP***n*** port (*rr* specifying the affected port, i.e. OC<M>/ STM<N>).

Behavior – OPT*n* Port

Loopback on OPT***n*** ports can be switched via the **`OPR-LPBK`** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to YES in case the *lpbktyp* = FACILITY
- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an ED-*rr* notification for an OPT***n*** port (*rr* specifying the affected port, i.e. OC<M>/ STM<N>)..

Behavior - EP155 Port

Loopback on EP***n*** ports can be switched via the **OPR-LPBK** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to YES in case the lpbktyp=FACILITY
- *fsflpbkstat* should be set to YES in case the lpbktyp=FSF.

This update is reported by an ED-rr notification for an EP155 port (*rr* specifying the affected port STM1E).

In case of a EP155 protection group the command is only applicable for the ports of the worker unit. Reason: Ports of the protection card are removed by creation of the protection group.

Behavior – Asynchronous port

Loopback on asynchronous ports can be switched via the **OPR-LPBK** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to YES in case the *lpbktyp* = FACILITY
- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an **ED-rr** notification for an asynchronous port (*rr* specifying the affected port). Not all types of asynchronous ports support loopbacks. See the modifier range for supported asynchronous ports.

Behavior - EC1 Port

Loopback on ports can be switched via the **OPR-LPBK** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port are updated by this command:

- *lpbk* should be set to YES in case the *lpbktyp* = FACILITY
- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an ED-EC1 notification for the port.

Behavior – GE1 Port

Loopback on GE1 ports can be switched via the **OPR-LPBK** command.

The attribute *lpbk* and *fsflpbkstat* of the entity port are updated by this command:

- *lpbk* should be set to YES in case the *lpbktyp* = FACILITY
- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an ED-EPORT notification for a GE***n*** port (only *fsflpbkstat*).

Behavior – Any Port on GE10PL1

Loopback on all successfully provisioned ports of a GE10PL1 pack can be switched via the **OPR-LPBK** command.

The parameter *fsflpbkstat* of the entity EPORT is updated by this command:

- *fsflpbkstat* should be set to YES in case the lpbktyp=FSF.

This update is reported by an ED-EPORT notification for the related EPORT (only for the parameter fsflpbkstat). Any attempts to change the parameter lpbktyp to the value FACILITY is rejected. The term ″any port″ refers to ports of both transmission rates, 1 GE and 10 GE.

Behavior for ONNS

Facility loopbacks are not allowed when portClass = EDGE or INNI.

Behavior FS - NS, NS - FS

Switching directly from an existing near-side loopback (far-side loopback) to a far-side loopback (near-side loopback) is not possible (i.e. is denied). Instead the existing loopback has to be released prior to establish another one.

Example 1

Provisioning a near-side facility loopback:

OPR-LPBK-STM16:UNITE-NE:1-1-#-#-1-1:123456:,,,FACILITY;

REPT DBCHG

ED-STM16:1-1-#-#-1-1::lpbk=YES;

Example 2

Provioning a far-side facility loopback:

OPR-LPBK-OC192:UNITE-NE:1-1-#-#-4-1:123456:,,,FSF;

REPT DBCHG

ED-OC192:1-1-#-#-4-1::fsflpbkstat=YES;

Updated affected entities

Apart from the port entity no further entities are affected by the **OPR-LPBK** command. Updates of the port entities are reported via ED-*rr* (OP***n*** port) or ED-EPORT (GE***n*** port) notifications.

Example input/output

Example:

Suppose for the port 1 of the pack in slot 01 a far-side facility loopback is set:

```
OPR-LPBK-STM16:UNITE-NE:1-1-#-#-1-1:123456::,,,FSF;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Port is out-of-service

Loopback command processing verifies that the affected port is in out-of-service state, i.e. *servicecond* = OUTOFSERVICE.

Error code: `SNVS – Status, Not in Valid State, Port is not out-of-service`

Validation rule: *modifier* consistent with given port-rate

Loopback command processing verifies that the *modifier* parameter is consistent with the rate of the port (given by the *aid* parameter). E.g. for an OP2G5 only the *modifier* values STM16 or OC48 (depending on the port's interface standard) are allowed, for an OP10 only *modifier* values STM64 or OC192 (depending on the port's interface standard) are allowed etc.

Error code:

```
IDNC – Input Data Not Consistent, modifier is inconsistent with
existing port rate
```

Validation rule: *modifier* consistent with given port's interface standard

Loopback command processing verifies that the *modifier* parameter is consistent with the interface standard of the port (this port given by the aid parameter). E.g. for an SDH port only the *modifier* values for STM16 or STM64 or STM256 are valid, for a SONET port only the *modifier* values OC48 or OC192 or OC768 are valid.

Error code:

```
ICNV – Input Command Not Valid, Interface Standard does not match
```

Validation rule: *modifier* consistent with the related circuit pack type

Loopback command processing shall verify in case that the modifier parameter does not distinguish various entities by transmission rate or by interface standard that the addressed entity belongs to the correct pack type. E.g. if the modifier value 1GE has been sent for an 1 GE port on the GE10PL1 pack or if the modifier value 1GEPL has been sent for a port on the GE1 pack, then the command is rejected.

Error code:

```
IDNC – Input, Data Not Consistent, modifier is inconsistent with the
related circuit pack type
```

Validation rule: Inconsistent facility loopback types on port pairs

The command is denied, if a near side or far side facility loopback is switched on an STM1E, STM1, STM4, OC3 or OC12 port, while on the other port of the same port pair a loopback of the other type already exists. A port pair is defined as 2 ports with consecutive port numbers n and n+1, where n can be any of the odd port numbers residing on the pack.

Error code:
`SNVS – Status, Not in Valid State, Facility loopback with different type exists at neighbor port`

Validation rule: Loopback must not already exist

Loopback command processing verifies that the loopback command can not be applied to an already existing loopback. The request to establish a near-side (far-side) loopback will be rejected if a near-side (far-side) loopback for the effected port already exists.

Error code: `SAOP – Status Already Operated, loopback already exists`

Validation rule: For ports on GE*n* packs only far-side loopback allowed

Loopback command processing shall verify that for an EPort only a far-side facility loopback (lpbktyp=FSF) can be switched.

Error code: `IDNV – Input Data Not Valid, Invalid loopback for EPort`

Validation rule: No change of loopback allowed

Switching directly from an existing near-side loopback (far-side loopback) to a far-side loopback (near-side loopback) is not possible (i.e. is denied).

Error code:
`SNVS – Status, Not in Valid State, Trial to change loopback type`

Validation rule: No loopbacks on ports of *portClass* INNI

Loopback command processing shall verify that the port has not *portClass* INNI.

Error code:
`SNVS – Status, Not in Valid State, Port must not be of portClass INNI`

Validation rule: No loopbacks on ports of *portClass* UNI

Loopback command processing shall verify that the port has not *portClass* UNI.

Error code:
`SNVS – Status, Not in Valid State, Port must not be of portClass UNI`

Validation rule: No loopbacks for PTF on LOXC allowed

Loopback command processing shall deny loopback requests for high-order path-termination functions residing on the LOXC pack.

Error code: `IDNV – Input, Data Not Valid, Invalid loopback for PTF on LOXC`

Related TL1 messages

- "RLS-LPBK: Release loopback (facility)" (p. 3-2198)
- "RTRV-LPBK (facility): Retrieve loopback (facility)" (p. 3-2511)

References

GR-199-CORE

OPR-LPBK notification (facility)

There will be no **`OPR-LPBK`** notification because the changes (for the parameter *lpbk* and *fsflpbkstat*) caused by an OPR-LPBK command will be notified via an ED-*rr* notification for an OP***n*** / EP***n*** / OPT***n*** port (*rr* specifying the affected port i.e. OC<M>/STM<N>, OC<M>T/STM<N>T or via an ED-EPORT notification for a GE***n*** port (*fsflpbkstat* only).

OPR-MAINT-ONNS: Maintenance actions for the ONNS application

The **OPR-MAINT-ONNS** command can be used to execute specific maintenance actions for the ONNS application. Such maintenance actions are, for example, removal of orphaned entities like cross-connections or cross-connection legs.

The type of maintenance action to be executed is specified by the parameter *reqType*. Depending on the value of parameter *reqType*, additional parameters in the spec block need to be specified.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P5

Priority: 1

Abortable: No

Input format

Command syntax: **OPR-MAINT-ONNS**:*srcTid*:*srcAid*:*ctag*:*reqType*:*spec_block*;

Input parameters

Command parameters

1. **srcTid**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **ctag** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

4. **`reqType`**

   This parameter indicates the maintenance operation that is to be performed.
   Type: Alphanumeric value set.

   Range:

   - REMLEG: removes a cross-connection leg from a “bridged” cross-connection of class NN. The cross-connection affected and the leg to be removed are specified via spec block parameter *pathID* or *connPtID*.
   - REMLEG_RESV: removes a cross-connection leg from a “bridged” cross-connection of class NN and installs an M:N reservation on the iNNI tributary the leg has been removed from. The cross-connection affected and the leg to be removed are specified via spec block parameter *pathID* or *connPtID*.

     Note that option *REMLEG_RESV* normally only makes sense if the leg to be removed is specified by the *pathID*.

Spec block parameters

1. **`pathID`**

   The PathID of the 1st path that was created.
   Type: character string of length up to 20.
   Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255) (Index: Numeric string 0-65535).
   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x
   Example: 29.13.196.203-65111
   Default: None.

   In case a value for *pathID* is given, the requested operation will only be applied to the leg having this path ID assigned only under the following conditions:

   - No value for parameter *connPtID* is additionally given
   - This path ID is assigned to a leg of a “bridged” cross-connection of class NN
   - This path ID is different from the path ID assigned to the second leg of the affected “bridged” cross-connection
   - This path ID is different from an all-zero path ID

     In this case it is recommended to execute an XC recovery or protection switch of the active XC in the affected node in order to restart the ONNS connection state machine, as there is a certain possibility for the affected ONNS connection for being in a state the state machine is not able to handle any more.

2. **`connPtID`**

   ONNS connection point identifier for the ingress tributary ID of the leg to be removed from a “bridged” cross-connection.
   It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
   Type: Dash-separated list of 5 numeric strings.

   Range:

   - Bay/Frame #1
   - Shelf #1
   - Slot/Pack #1…39

- Port #1…16
- Tributary #1…128

Default: None.

In case a value for *connPtID* is given, the requested operation is applied to the leg identified by the specified *connPtID* only under the following conditions:

- No value for parameter *pathID* is additionally given
- The specified connPt identifies a leg of a "bridged" cross-connection of class NN
- The *pathID* associated to the specified leg is the all-zero *pathID* or is equal to the *pathID* associated to the second leg of the "bridged" cross-connection of class NN.

Output format

If the **OPR-MAINT-ONNS** request is accepted by the system, the following normal completion response format below is returned:

```
   srcTid date time
M  ctag COMPLD
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *conpletion code* and their syntax are included in the section "Output parameters" of the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) command.

Example input/output

The following example shows a successful completion of an **OPR-MAINT-ONNS** command for removal of an orphaned cross-connection leg specified by the associated *connPt* ID:

```
OPR-MAINT-ONNS:LT-ONNS-789:snn:123456:REMLEG:
connPtID=1-1-3-2-12;

   LT-ONNS-789 2008-05-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a successful completion of an **OPR-MAINT-ONNS** command for removal of an orphaned cross-connection leg with installation of M:N reservation, where the leg is specified by the path ID:

```
OPR-MAINT-ONNS:LT-ONNS-789:snn:123456:REMLEG_RESV:
```

```
pathID=135.14.196.203-32165;

   LT-ONNS-789 2008-05-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer also to the section "Error responses" of "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698). The error responses listed there also apply to the **OPR-MAINT-ONNS** command.

Validation rule: OPR-MAINT-ONNS with *reqType* = REMLEG or REMLEG_RESV not allowed because of specified connection is not valid

The **OPR-MAINT-ONNS** command with *reqType* = REMLEG or REMLEG_RESV is rejected with the following error response, in case the connection addressed does not have class NN or is not a "brdiged" cross-connection:

Error code: `E700 - ICID - Invalid Connection IDentification`

Validation rule: OPR-MAINT-ONNS with *reqType* = REMLEG or REMLEG_RESV not allowed because of both pathID and connPtID specified

The **OPR-MAINT-ONNS** command with *reqType* = REMLEG or REMLEG_RESV is rejected with the following error response, in case of both *pathID* and *connPtID* for leg identification are specified in the command:

Error code: `E701 - DLID - Duplicate Leg IDentification`

Validation rule: OPR-MAINT-ONNS with *reqType* = REMLEG or REMLEG_RESV not allowed because of invalid leg identification via pathID or connPtID

The **OPR-MAINT-ONNS** command with *reqType* = REMLEG or REMLEG_RESV is rejected with the following error response, in case of the leg identification via *pathID* or *connPtID* is not supported:

Error code: `E702 - ULID - Unsupported Leg IDentification`

Related TL1 messages

- "AUD: Perform audit data" (p. 3-46)
- "RTRV-AUDLOG: Retrieve audit result log for ONNS connection data" (p. 3-2263)

OPR-PROTNSW: Operate protection switch

The **OPR-PROTNSW** command allows to switch one or more existing protection group instances of any protection group type. Different identification methods are supported for the different protection types.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

1+1 equipment group

Command syntax: **OPR-PROTNSW**:*tid*:*aid*:*ctag*::*protype*:*spec_block*;

Multiplex section/line protection group

Command syntax: **OPR-PROTNSW**:*tid*:*aid*:*ctag*::*protype*:*spec_block;*

2-/4-fibre MS-SPRing/BLSR protection group

Command syntax: **OPR-PROTNSW**:*tid*:*[aid]*:*ctag*::*protype*,*[rid]*:*spec_block*;

Path protection group

Command syntax: **OPR-PROTNSW**:*tid*:*[aid]*:*ctag*::*protype, ,[ppgname][,aid1]*:*spec_block*;

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.

> **Important!** The use of *ppgname* as input parameter with alias function is not yet supported. Address path-protected cross-connections individually via *Navis*® Optical EMS/CIT GUI using their AIDs.

> **Important!** If a OPR-PROTNSW command is executed on an MSP 1+1 optimized while the actual protection state does not allow the specified protection switch, then the command is denied with the wrong error code:
> `SROF – Status, Requested Operation Failed..`

The expected error response is:

```
SNVS – Status, Not in Valid State, switch command not allowed for
actual protection state
```

Input parameters

Command parameters (1+1 equipment protection group)

1. **tid** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   Description: The parameter specifies the aid of 1+1 equipment protection group.
   Type: AID
   Range: SwitchAndTmgProtGrpAID (for higher-order switch fabirc and timing group) or CtlProtGroupAID (for system controller protection group) or EP155ProtGroupAID (for EP155 protection group) or EP51ProtGroupAID (for electrical 51 MBit/s groups) or LOXCProtGroupID (for lower-order switch fabric groups) or TMUXProtGroupAID (for TransMUX groups) or ES64UProtGroupAID (for ES64U groups).
   Default: No default
3. **ctag** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **protype** – Protection Type
   The parameter specifies the type of the 1+1 equipment protection group for retrieving the information.
   Type: Alphanumeric value set.
   Range:
   - EQPTSWFBR, if *aid* is SwitchAndTmgProtGrpAID;
   - EQPTCTL, if *aid* is CtlProtGroupAID;
   - EQPTEP155, if *aid* is EP155ProtGroupAID;
   - EQPTEP51, if *aid* is EP51ProtGroupAID;
   - EQPTLOXC, if *aid* is LOXCProtGroupAID.
   - EQPTTMUX, if *aid* is TMUXProtGroupAID.
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

Command parameters (multiplex section/line protection group)

1. **tid** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port aid being a member of the group.
   Type: AID
   Range: OnePlusOne_ProtGrpAID (not ranged) or NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged).

3. **`ctag`** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **`protype`** – Protection Type
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later)
   Type: Value set.
   Range: 1+1 or 1xNOPT

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **`ctag`** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F
5. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   If *aid* is specified, then *rid* can be omitted.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.

Command parameters (Path protection group)

1. **`tid`** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies a path protection group aid or an input tributary aid being a member of the group in combination with *ppgname*.
   The parameter specifies an input tributary *aid* when used in combination with parameter *aid1*.
   Option: If *ppgname* is specified, then *aid* can be omitted or used as an input tributary aid. The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of ppgname is not recommended. For low order path protections, the aid is mandatory since the *ppgname* is not supported for those protection groups. For such protection groups, always a path protection group AID has to be specified.
   Option: If parameters aid specifies a HO tributary on a port of port class EDGE and this tributary is combined with another HO tributary within an OUC, then the command shall operate on all protection groups of type PATHDRI identified by these two HO tributaries, if existing. If aid specifies a HO tributary on a port of port class INNI, the command is rejected.
   Option: When parameter aid is used in combination with parameter *aid1*, then the command shall operate on all protection groups of type PATHDRI having the AIDs given in parameters *aid* and *aid1* as input tributaries. If *aid* or *aid1* specifies a tributary on a port of port class INNI, the command is rejected.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.
3. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: PATHDRI
4. **`ppgname`** – Path protection group Name
   In particular follow the reference to check options and mandatory omissions associated with this parameter.
   The parameter specifies the path protection group name. Storage and retrieval of the ppgname is case sensitive, but actual use of the ppgname to determine the corresponding protection group is case insensitive.
   Option for high order path protection groups: For high order path protections, if aid is specified, then *ppgname* can be omitted. For low order path protections, the *ppgname* is omitted as not being supported at all. If an input value is specified for ppgname and none is specified for aid, then RTRV-PROTN-GRP operates on all

protection groups of type PATHDRI that have the specified path protection group name. If input values are specified for *aid* and *ppgname* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified *aid*, *ppgname* pair. The command will complete successfully if there is no protection group described by *aid*, *ppgname*, but no data will be returned.
The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
Type: text string
Range: 1 to 24 characters

5. **`aid1`** – Access identifier 1
   Together with parameter *aid* this parameter specifies a pair of tributaries in order to execute switches to emulated selector bridges in one command.
   Option: If input values are specified for *aid* and *aid1* then OPR-PROTNSW operates on all protection groups of type PATHDRI that have these specified tributaries as inputs. If input values are ecified for *aid* and *aid1*and additionally a value is specified for *ppgname*, OPR-PROTNSW shall operate only on those protection groups that additionally match the ppgname value. *aid1* specifies a HO tributary in a port of port class INNI, the command is rejected.
   Type: AID
   Range: STS1NumAID. The use of wildcards is not allowed.

Command parameters (Constituent Path Protection Group)

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies a path protection group aid or an input tributary aid being a member of the group in combination with *ppgname*.
   Option: If *ppgname* is specified, then aid can be used as an input tributary aid. The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Type: AID
   Range: STS1NumAID, VCGNumAID
   The use of wild cards is not restricted for STS1NumAID and VCGNumAID. No wildcards are allowed in a GenLOTribAID.
   This range specifies all AIDs, which can be used to identify a constituent path protection group.
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

4. **`protype`** – Protection Type
   The parameter specifies the protection group type to be identified.
   Type: Alphanumeric value set.
   Range: CONSTITUENTPATH

5. **`ppgname`** – Path protection group Name
   The parameter specifies the path protection group name. Storage and retrieval of the ppgname is case sensitive, but actual use of the ppgname to determine the corresponding protection group is case insensitive.
   Option for high order path protection groups: For high order path protections, if aid is specified, then *ppgname* can be omitted. For low order path protections, the *ppgname* is omitted as not being supported at all. If an input value is specified for ppgname and none is specified for aid, then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified path protection group name. If input values are specified for *aid* and *ppgname* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified *aid*, *ppgname* pair. The command will complete successfully if there is no protection group described by *aid*, *ppgname*, but no data will be returned.
   The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Type: text string
   Range: 1 to 24 characters

Spec block parameters (1+1 equipment protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, sc*.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command; i.e. which side is active in terms of the member role within the protection group after the switch has been executed.

   The following values are supported:

   - WKG: Worker
   - PROTN: Protection

   Type: Alphanumeric value set.
   Range: WKG, PROTN

2. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.

The following values are supported:

- FRCD: Forced Switch
- MAN: Manual Switch.

Type: Alphanumeric value set.
Range: MAN, if protype is EQPTCTL; FRCD, MAN (for all other equipment protection group types).

Spec block parameters (multiplex section/line protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, sc*.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command; i.e. to which side the switch command shall applied to.
   The following values are supported:
   - WKG: Worker
   - PROTN: Protection

   The parameter will not be taken into consideration, if a LOCKOUT switch command is performed. Details see behavior description.
   Type: Alphanumeric value set.
   Range: WKG, PROTN.
2. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.
   The following values are supported:
   - LOCKOUT: Lockout
   - FRCD: Forced Switch
   - MAN: Manual Switch

   Type: Alphanumeric value set.
   Range: LOCKOUT, FRCD, MAN

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, destside, sc*, swtype.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command. The following values are supported: PROTN, Protection.
   Type: Alphanumeric value set.
   Range: PROTN.
2. **`destside`** – Destination Side

This parameter specifies the destination side to which the switch command is applied; i.e.

- it is EAST for the east ring side
- it is WEST for the west ring side.

Type: Alphanumeric value set.
Range: EAST, WEST.

3. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.

   The following values are supported:

   - LOCKOUT: Lockout
   - FRCD: Forced Switch
   - MAN: Manual Switch.

   Type: Alphanumeric value set.
   Range: LOCKOUT, FRCD, MAN.
4. **`swtype`** – Switch Type
   This parameter specifies what type of protection switching is used for the MS-SPRing/BLSR protection group. The value RING is used for loopback switching. The value SPAN can be used in case of span switching.
   Type: Alphanumeric value set.
   Range: RING, SPAN.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, destside, sc,* swtype.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command. The following values are supported: PROTN, Protection.
   Type: Alphanumeric value set.
   Range: PROTN
2. **`destside`** – Destination Side
   This parameter specifies the destination side to which the switch command is applied; i.e.

   - it is EAST for the east ring side
   - it is WEST for the west ring side.

   Type: Alphanumeric value set.
   Range: EAST, WEST.
3. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.

The following values are supported:

- LOCKOUT: Lockout
- FRCD: Forced Switch
- MAN: Manual Switch

Type: Alphanumeric value set.
Range: LOCKOUT, FRCD, MAN.

4. **`swtype`** – Switch Type
   This parameter specifies what type of protection switching is used for the MS-SPRing/BLSR protection group. The value RING is used for loopback switching (i.e. service route is looped back onto the protection route in the direction opposite the failure). The value SPAN is used in case of span switching (i.e. all traffic being carried on the worker span between an adjacent pair of nodes is switched to the protection span, between the same pair of nodes and on the same side of the ring).
   Type: Alphanumeric value set.
   Range: RING, SPAN.

Spec block parameters (path protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, sc*.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command; i.e. to which side the switch command is applied to.
   The following values are supported:
   - WKG: Worker
   - PROTN: Protection.

   Type: Alphanumeric value set.
   Range: WKG, PROTN
2. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.
   The following values are supported:
   - LOCKOUT:Lockout
   - FRCD: Forced Switch
   - MAN: Manual Switch.

   Type: Alphanumeric value set.
   Range: LOCKOUT, FRCD, MAN.

Spec block parameters (constituent path protection group)

See detailed behavior description w.r.t. allowed switch commands; i.e allowed value combinations for parameters *dest, sc*.

1. **`dest`** – Destination
   This parameter specifies the destination of the switch command; i.e. to which side the switch command is applied to.
   The following values are supported:
   - WKG: Worker
   - PROTN: Protection.

   Type: Alphanumeric value set.
   Range: WKG, PROTN
2. **`sc`** – Switch Command
   This parameter specifies the switch command to be executed by the protection state machine.
   The following values are supported:
   - LOCKOUT:Lockout
   - FRCD: Forced Switch
   - MAN: Manual Switch.

   Type: Alphanumeric value set.
   Range: LOCKOUT, FRCD, MAN.

Output format

After receiving the **`OPR-PROTNSW`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Detailed behavior description

The command switches the identified protection group. The switch command is processed according to the state machine behavior for the different protection group types. As consequence, the values of the following attributes of the state machine may be updated:

EQPTSWFBR: *actunit, swreq*

EQPTCTL: *actunit, swreq*

EQPTEP155: *actunit, swreq*

EQPTEP51: *actunit, swreq*

EQPTLOXC: *actunit, swreq*

EQPTTMUX: *actunit*, *swreq*

EQPTES64U: *actunit*, *swreq*

1+1/1:1: *actunit*, *swreq*

2F: *eswreq*, *wswreq*

4F: *eactunit*, *eswreq*, *wactunit*, *wswreq*,

PATHDRI: *actunit*, *actunitaid*, *swreq*

The update of the attributes is reported by means of a protection switch notification for the corresponding protection group type.

Behavior of path-protected cross-connections in Fixed and Adaptive Mode

If the protection type specified in the command is PATHDRI, then the command ispplied to all constituent members of the path protected cross-connection.

- If there are two or more constituent members, and the protection switch state of one or more constituent members does not allow the command execution, then the command is successfully completed. The system sends protection switch notifications only for those constituent members, where the command has been successfully executed.
- If there is only one constituent member, and the protection switch state does not allow the command execution, then the command is denied with the appropriate error response. This behavior is applicable for path protected cross-connections in fixed and adaptive mode. In fixed mode, there can be only one constituent member, and the command will be denied if it cannot be executed. Path protected pipe mode cross-connections with only one constituent member show the same behavior as path protections in fixed mode.

Identification of path protection groups

For PATHDRI, the protection switch applies to all constituents of the identified tributary. For this case, the following table summarizes the use of path protection group names as related to OPR-PROTNSW.

aid	ppgname	dest (note 2)	comment
not used	ppgname	WKG or PROTN	operation applies to all tributaries that are group members of the ppgname (note 1)
grp member (input) logical tributary AID	ppgname	WKG or PROTN	operation applies to all tributaries that are group members of the aid, ppgname subset (note 1)
path prot grp id (output logical tributary AID)	not used	WKG or PROTN	normal, non-aliasing target identification (note 1, note 2)

Note 1: group members are defined to be the working and protection input logical tributaries of the path protection group.

Note 2: the output logical tributary aid of an adjunct path-protected cross-connection is not a path prot grp id, only the output of a path-protected cross-connection is.

Allowed/supported switch commands for 1+1 Equipment Protection

The following table shows the allowed/supported value combinations for the parameters *sc*, *dest* w.r.t. OPR-PROTNSW for 1+1 equipment protection group.

sc	dest
FRCD	WKG
FRCD	PROTN
MAN	WKG
MAN	PROTN

Note 1: For the controller protection group instance only *sc* = MAN is applicable.

Behavior description for multiplex section/line protection

The destination parameter dest is noten taken in consideration, if a LOCKOUT command is performed.

This means for the different MSP types:

- 1:1/1+1 : allways the protection side is locked out and the working side carries the traffic. (By this (PROTN) was specified in the tables)
- 1+1 optimized: The actual protection state is frozen. The actual inactive side is locked out. (To make it also consistent to the other MSPs also (PROTN) was specified in the table).

Allowed/supported switch commands for 1:1 MSP Protection SDH

The following table shall show the allowed/supported value combinations for the parameters sc, dest, destside, swtype dest w.r.t. OPR-PROTNSW for 1:1 Multiplex Section/Line protection group for SDH mode:

sc	dest
FRCD	WKG
FRCD	PROTN
MAN	PROTN
LOCKOUT	(PROTN)

Allowed/supported switch commands for 1+1 MSP Protection

The following table shall show the allowed/supported value combinations for the parameters sc, dest w.r.t. OPR-PROTNSW for 1+1 Multiplex Section/Line protection group:

sc	dest
FRCD	WKG
FRCD	PROTN
MAN	WKG
MAN	PROTN
LOCKOUT	(PROTN)

Allowed/supported switch commands for 1+1 Optimized MSP

The following table shall show the allowed/supported value combinations for the parameters sc, dest w.r.t. OPR-PROTNSW for 1+1 optimized Multiplex Section protection group:

sc	dest
FRCD	WKG
FRCD	PROTN
LOCKOUT	(PROTN)

Allowed/supported switch commands for 2-fibre MS-SPRing/BLSR

The following table shows the allowed/supported value combinations for the parameters *sc, dest, destside, swtype* w.r.t. OPR-PROTNSW for 2-fibre MS-SPRing/BLSR protection groups.

Case	sc	dest	destside	swtype
LP-S	LOCKOUT	PROTN	EAST or WEST	SPAN
FS-R	FRCD	not used/ omitted	EAST or WEST	RING
MS-R	MAN	not used/ omitted	EAST or WEST	RING

Allowed/supported switch commands for 4-fibre MS-SPRing/BLSR

The following table shall show the allowed/supported value combinations for the parameters sc, dest, destside, swtype w.r.t. OPR-PROTNSW for 4-fibre MS-SPRing/BLSR protection groups.

Case	sc	dest	destside	swtype
LP-S	LOCKOUT	PROTN	EAST or WEST	SPAN
FS-R	FRCD	not used/ omitted	EAST or WEST	RING
FS-S	FRCD	not used/ omitted	EAST or WEST	SPAN
MS-R	MAN	not used/ omitted	EAST or WEST	RING
MS-S	MAN	not used/ omitted	EAST or WEST	SPAN

Allowed/supported switch commands for path protection group

The following table shows the allowed/supported value combinations for the parameters *sc*, *dest* w.r.t. OPR-PROTNSW for path protection groups.

Dependency of allowed/supported switch request were simplified in order to:

- comply with new GR-1400 (01/99)
- harmonize with 1+1 Multiplex section/Line protection group.

sc	dest
LOCKOUT	PROTN
FRCD	WKG
FRCD	PROTN
MAN	WKG
MAN	PROTN

Example input/output

1+1 Equipment protection group

The following example shows an successful ’manual switch to protection’ command for a 1+1 equipment protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-estgrp:123456::EQPTSWFBR:sc=MAN,dest=PROTN;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Multiplex section/line protection group 1+1

The following example shows an successful ’forced switch to worker’ command for a 1+1 multiplex section/line protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-o001:123456::1+1:sc=FRCD,dest=WKG;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Multiplex section/line protection group 1:1 (SDH)

The following example shows an successful ’manual switch to protection’ command for a 1:1 multiplex section/line protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-o001:123456::1xNOPT:sc=MAN,dest=PROTN;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Multiplex section/line protection group 1:1 optimized

The following example shows an successful lockout command for a 1+1 optimized multiplex section/line protection group: Please keep in mind, that the *dest* parameter is not taken in consideration:

```
OPR-PROTNSW:LT-UNITE:1-1-o001:123456::1+1:sc=LOCKOUT,dest=PROTN;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

2 fibre MS-SPRing/BLSR protection group

The following example shows an successful ’manual ring switch’ command for the east side of a 2F MS-SPRing/BLSR protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-t001:123456::2F:sc=MAN,swtype=RING,destside=EAST;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

4 fibre MS-SPRing/BLSR protection group

The following example shows an successful ’manual span switch’ command for the west side of a 4F MS-SPRing/BLSR protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-f001:123456::4F:sc=MAN,swtype=SPAN,destside=WEST;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
```

```
;
```

Path protection group

The following example shows an successful ’lockout of protection’ command for a path protection group:

```
OPR-PROTNSW:LT-UNITE:1-1-#-#-10-1-25:123456::PATHDRI:sc=LOCKOUT,dest=PROTN;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a successful ’lockout of protection’ command to both path protection groups forming an emulated selector bridge on input tributaries 1-1-#-#-8-1-25 and 1-1-#-#-6-1-25:

```
OPR-PROTNSW:LT-UNITE:1-1-#-#-8-1-25:123456::PATHDRI,,,
1-1-#-#-6-1-25:sc=LOCKOUT,dest=PROTN;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses (general)

Validation rule: complete switch command

The switch command (combination of *sc*, *dest*, *destside*, *sctype*) is complete; i.e. all needed parameters are present.

Error code: `IDNV – Input, Data Not Valid, incomplete switch command`

Validation rule: supported switch command

The switch command (combination of *sc*, *dest*, *destside*, *sctype*) is supported by the specified protection group type.

Error code: `IDNV – Input, Data Not Valid, invalid switch command`

Validation rule: allowed switch command in that state

The specified switch command is allowed in the actual protection state.

Error code:
`SNVS – Status, Not in Valid State, switch command not allowed for actual protection state`

Validation rule: consistent aid and protype

The specified *aid* is consistent with the specified *protype*.

Error code: `IIAC – Input, Invalid ACcess identifier, incorrect syntax`

Error responses (1+1 Equipment protection group)

Validation rule: Clock not in warming-up status (Switching Protection Group)

The clock of the destination as requested in the manual switch command is not in the 'warming-up' state.

Error code:
`SNVS – Status, Not in Valid State, clock of destination XC is in warming-up state, please wait some minutes`

Validation rule: No forced switch to empty XC slot (Switching Protection Group)

The system shall not allow to execute a forced switch on a Switching Protection Group to an empty XC slot.

Error code: `ENEQ – Equipage, Not EQuipped, XC destination slot empty`

Validation rule: No forced switch to unusable XC slot (Switching Protection Group)

The system shall not allow to execute a forced switch on a Switching Protection Group to a XC slot which has an open latch or a power failure.

Error code: `SNVS – Status, Not in Valid State, XC destination not usable`

Validation rule: No manual protection switch if the switchover leads to traffic impairments

The system shall not allow to execute a manual switch on a Switching or LOXC Protection Group with a combination of XC circuit pack types which leads to traffic impairments in the case of protection switches.

Error code:
`SNVS – Status, Not in Valid State, Protection switch with current combination of circuit pack types is traffic affecting`

Validation rule: No forced switch to empty LOXC slot (LOXC Protection Group)

The system shall not allow to execute a forced switch on an LOXC Protection Group to an empty LOXC slot.

Error code: `ENEQ – Equipage, Not EQuipped, LOXC destination slot empty`

Validation rule: No forced switch to unusable LOXC slot (LOXC Protection Group)

The system shall not allow to execute a forced switch on an LOXC Protection Group to an LOXC slot which has an open latch or a power failure.

Error code: `SNVS – Status, Not in Valid State, LOXC destination not usable`

Validation rule: No forced switch to empty slot (TransMUX Protection Group)

The system does not allow to execute a forced switch on a TransMUX protection group to an empty slot.

Error code: `ENEQ – Equipage, Not EQuipped, TMUX destination slot empty`

Validation rule: No forced switch to unusable TransMUX pack (TransMUX Protection Group)

The system does not allow to execute a forced switch on a TransMUX protection group to a TransMUX pack which has an open latch or a power failure.

Error code: `SNVS – Status, Not in Valid State, TMUX destination not usable`

Validation rule: No forced switch to empty slot (ES64U Protection Group)

The system does not allow to execute a forced switch on an ES64U protection group to an empty slot.

Error code: `ENEQ – Equipage, Not EQuipped, ES64U destination slot empty`

Validation rule: No forced switch to unusable ES64U pack (ES64U Protection Group)

The system does not allow to execute a forced switch on a ES64U protection group to a ES64U pack which has an open latch or a power failure.

Error code:

`SNVS – Status, Not in Valid State, ES64U destination not usable`

Error responses (MS/Line protection group)

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Validation rule: MSP type 1:1 does not allow manual switch requests to worker

The system shall reject an OPR command to select the worker port as active port (actunit=WKR) if the current protection architecture is 1:1 (protype = 1x1). This applies only to manual switch requests.

Error code:

```
SNVS – Status, Not in Valid State, Manual Switch to Worker for
Protection Type 1x1 not allowed
```

Validation rule: MSP type 1+1_optimized does not allow manual switch commands

The system shall reject an OPR command for manual switch requests if the operation is ’optimized’.

Error code:

```
SNVS – Status, Not in Valid State, Manual Switch for Protection Type
1+1_Optimized not allowed
```

Error responses (2-/4-fibre MS-SPRing/BLSR protection group)

Validation rule: consistent *aid* and *rid*

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Validation rule: APS switch time constraint

The switch command is executed by the protection state machine using the APS within time constraints

Error code: `SPFA – Status, Protection unit FAiled, APS did not succeed`

Error responses (Path protection group)

Validation rule: Protection Group must exist

If the parameter *aid*, or the combination of the parameters *aid* and *ppgname* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Boundary consistency for constituent members

When the protection type specified in the command is CONSTITUENTPATH and the constituent member of the path-protected pipe cross-connection is a concatenated signal, then the command is only executed if the specified tributary AID is the first tributary of the concatenated signal.

Error code:
`IENE – Input, Entity Not Exists, constituent member does not exist`

Validation rule: Protection Group must exist

If the combination of the parameters *aid* and *aid1* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Validation rule: *ppgname* and low order path protections

If a *ppgname* is given as input parameter while the *aid* references a low order tributary, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, ppgname not supported for this transmission rate`

Validation rule: Access via tributary on INNI port

If in input parameter *aid* or *aid1* for access via input or output a tributary is given located on a port of port class INNI, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, No Access via AID located on INNI port`

Related TL1 messages

- “RTRV-PROTN-GRP: Retrieve protection group” (p. 3-2759)
- “ED-PROTN-GRP: Edit protection group” (p. 3-803)
- “ENT-PROTN-GRP: Enter protection group” (p. 3-1745)
- “ENT-CRS: Enter cross-connection” (p. 3-1464)
- “RLS-PROTNSW: Release protection switch” (p. 3-2203).

OPR-SYNCNSW: Operate synchronization switch

The **OPR-SYNCNSW** command can be initiated by user to request the NE to operate manual or forced synchronization reference protection switch or external timing output protection switch (for SDH only), to lockout synchronization reference or external output timing (for SDH only) switching or to request a change from the provisioned synchronization mode to forced holdover state.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

Command syntax: **OPR-SYNCNSW**-*modifier*:*tid*::*ctag*::*sc[,dest]*;

Input parameters

Command parameters

1. **modifier** – modifier
   This parameter indicates the management system on which the switch request acts. The *modifier* REF specifies the assigned timing reference switch. The *modifier* MOD specifies the clock mode switch of the system clock. The *modifier* OUTREF specifies the external timing output switch (for SDH only).
   Type: Alphanumeric value set.
   Range:
   - REF
   - MOD
   - OUTREF
2. **tid** – target identifier
   See definition of this parameter in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – correlation tag
   See definition of this parameter in "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **sc** – switch command
   Note: When entering an OPR-SYNCSW-OUTREF command this parameter must be specified in uppercase letters, otherwise the command is rejected with error code: `SNVS /* Status, Not in Valid State, parameter required */`
   This parameter defines the type of switch action that is initiated on the timing reference or mode.

The following values are valid depending on parameter *modifier*.

- *modifier* REF: LOCKOUT – The reference which is addressed by the *dest* parameter, will be locked out by the user and cannot be used as a timing reference signal until the lockout is cleared. FRCD – A forced switch is initiated. It forces selection of the specified reference determined by the *dest* parameter. MAN – A manual switch is initiated to the specified reference determined by the *dest* parameter. CLRWTR – The wait to restore period of the reference which is addressed by the *dest* parameter will be terminated. The signal status will return to normal.
- *modifier* MOD: FRCD – A forced switch is initiated. It forces system timing from the locked mode to the holdover mode (Clock Backup Request).
- *modifier* OUTREF (for SDH only): LOCKOUT – The reference which is addressed by the *dest* parameter, will be locked out by the user and cannot be used for external timing output until the lockout is cleared. FRCD – A forced switch for external timing output is initiated. It forces selection of the specified reference determined by the *dest* parameter. MAN – A manual switch for the external timing output is initiated to the specified reference determined by the the *dest* parameter.

LOCKOUT has the highest priority, followed by FRCD, then MAN.
Type: Alphanumeric value set.

Range:

- LOCKOUT
- FRCD
- MAN
- CLRWTR.

Default: Not applicable.

5. **`dest`** – destination
   This parameter specifies the assigned timing reference identifier destination. This is the list of timing references available to the system clock or to the timing link switch of the external output timing. This parameter is valid and required for the REF and OUTREF modifiers.
   Type: AID

   Range:

   - *modifier* REF: External_timing_ref_AID, Line_timing_ref_AID
   - *modifier* OUTREF (for SDH only): Line_timing_ref_AID

   Default: Not applicable.

Output format

After receiving the **`OPR-SYNCNSW`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
```

;

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **OPR-SYNCNSW** command generates no **REPT DBCHG** Report Database Change notification. The **OPR-SYNCNSW** command returns a REPT EVT Report Event notification in consequence of a protection switch notification.

Clock backup request

Forced Holdover: The forced switch of system clock to holdover external request causes the system clock to be switched to holdover regardless of the state of the timing references. The following parameter is affected by the clock mode switch request Forced Holdover: – clkmod_stat = FORCED-HOLDOVER.

System timing reference or output timing switch request

Forced Switch: The forced switch to timing reference external switch request causes the system to be switched to the selected timing reference regardless of the state of that timing reference. While in the Forced Switch state the system may not switch the active timing reference, neither automatically nor by means of the Forced Switch or Manual Switch command.

Manual Switch: The manual switch to timing reference external request causes the system to be switched to the selected timing reference. The system leaves the Manual Switch state by means of the Clear, Lockout or Forced Switch commands. While in the Manual Switch state the system may switch the active timing reference automatically if required for protection switching, the system leaves the Manual Switch state due to automatic switching.

Lockout: The Lockout of reference external request is provided to allow manual disabling of individual references. While in the Lockout state the system may not switch the active timing reference to the reference, which is locked out.

The following parameters are affected by the system timing reference switch request:

- active_ref = [External_timing_ref_AID, Line_timing_ref_AID]
- refsw_stat = [FORCED, MANUAL]
- ref_lock = LOCKOUT
- ref_stat = NORMAL.

The following parameters are affected by the external output timing switch request:

- active_oref_stat = Line_timing_ref_AID
- orefswr_stat = [FORCED, MANUAL]
- sclkref_lockstat = LOCKOUT.

Example input/output

The following is an example of the **OPR-SYNCNSW** command for System Timing with *modifier* MOD:

```
OPR-SYNCNSW-MOD:LT-UNITE::123456::FRCD;

   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **OPR-SYNCNSW** command for System Timing with *modifier* REF:

```
OPR-SYNCNSW-REF:LT-UNITE::123456::MAN,EXTREF1;

   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **OPR-SYNCNSW** command for Output Timing with *modifier* OUTREF:

```
OPR-SYNCNSW-OUTREF:LT-UNITE::123456::LOCKOUT,LINE3;

   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Parameter sc or dest is missing

If the system receives a OPR-SYNCNSW command with modifiers REF or OUTREF and one of the parameters *sc* or *dest* are missing, the system responds with:

Error code: `SNVS – Status, Not in Valid State, parameter required`

Validation rule: Invalid switch command

If the system receives a command with a *sc* value that is not supported, the system responds with:

Error code: `IDNV – Input, Data Not Valid, invalid sc`

Validation rule: Reference switching during lockout or disabled state

If the system receives a switch command to reference which is locked out or disabled, the system responds with:

Error code: `SNVS – Status, Not in Valid State, reference unavailable`

Validation rule: Clear wait to restore timer not applicable

If the system receives a clear request for a wait to restore timer while the reference is not in wait to restore state, the system responds with:

Error code: `SNVS – Status, Not in Valid State, clear request impossible`

Validation rule: Parameter not supported for modifier

If the system receives a OPR-SYNCNSW command with *modifier* MOD together with not supported parameter *dest*, the system responds with:

Error code: `IDNV – Input, Data Not Valid, parameter not required`

Validation rule: Clock backup switch request when clock mode is free running

If the system receives a OPR-SYNCNSW command with *modifier* MOD while the provisioned synchronization mode is free running, the system responds with:

Error code: `SSRD – Status, Switch Request Denied, clock mode inconsistent`

Validation rule: No change of the request state, request not practicable

If the system receives a switch command that results in no change to the switch request state or the active unit state, the system responds with:

Error code: `SNVS – Status, Not in Valid State, request not practicable`

Validation rule: Forced or manual switch request while reference is in forced state

If the system receives a manual or forced switch request to a reference which is already in forced state, the system responds with:

Error code: `SNVS – Status, Not in Valid State, request not practicable`

Related TL1 messages

- "RLS-SYNCNSW: Release synchronization switch" (p. 3-2215)
- "SET-SYNCN: Set synchronization" (p. 3-3158)
- "RTRV-SYNCN: Retrieve synchronization" (p. 3-2955)

References

GR-833-CORE

OPR-VCGTRIB: Operate virtual concatenation group tributary

The **OPR-VCGTRIB** command allows to preempt a running wait-to-restore at a VCG tributary being memeber of a LCAS protection scheme. The wait-to-restore status is part of the LCAS protection scheme.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **OPR-VCGTRIB-modifier**:*tid:aid:ctag::sc;*

Input parameters

Command parameters

1. **modifier**
   This parameter soecifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: STS, VC
2. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   The access identifier specifies the VCG tributary whose attributes are to be modified.
   Type: AID

   Range: VCGNumAID as defined in AID structure. The values

   - *all* for Simple_VCGNumAID
   - *vall* for Simple_VCGAID
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   is excluded.

4. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
5. **sc** – switch command
   The parameter *sc* specifies the switch action applied to the VCG tributary.
   Type: Alphanumeric value set.
   Range: CLRWRT.

Output format

If the NE fully complies with the **OPR-VCGTRIB** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The command shall preempt the wait-to-restore timer of the VCG tributary, addressed. State changes caused by this preemption are reported according an autonomous ED-VCGTRIB notification for this VCGTRIB for the parameter *inmbst*.

Example input/output

```
OPR-VCGTRIB-VC:LT-UNITE:1-1-#-#-1-v1-2:123456::CLRWTR;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **OPR-VCGTRIB** command.

Validation rule: Check modifier against circuit pack type

If the command is applied with modifier = VC or STS to a VCG Tributary on a Transparent pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: wait-to-restore is not pending

Check that addressed tributary is in state wait-to-restore if parameter *sc* = CLRWTR is applied.

Error code:

`SNVS – Status, Not in Valid State, Tributary is not in state wait-to-restore`

Related TL1 messages

- "RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109)
- "ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)

Protection switch notification

Output format

General

The following syntax is used for: 1+1 equipment, 1+1 multiplex section/line, 1:N multiplex section/line, 2-/4-fibre MS-SPRing/BLSR, path protection group.

1+1 equipment protection includes LOXC, TransMUX, and ES64U equipment protection groups.

Command syntax: **`REPT EVT`** *modifier aid*:*condtype*,*condeff*,*ocrdat*,*ocrtm*:*conddescr;*

1+1 equipment protection group

The REPT EVT Notification is generated if the swreq (switch request state) attribute is changed due to an automatic switch or a switch command. In addition, a REPT SW Notification is generated if the actunit (active unite state) attribute is changed due to an automatic switch or a switch command.

If both, actunit and swreq are changed due to an automatic switch or a switch command, then one REPT SW Notification and one REPT EVT Notification is generated.

Output parameters

Command parameters (1+1 equipment protection group, REPT EVT)

1. **`modifier`** – Modifier
   This parameter specifies the value ’COM’ as an event that applies to the whole NE.
   Type: Alphanumeric value set.
   Range: COM.
2. **`aid`** – Access Identifier
   Description: The parameter specifies the aid of 1+1 equipment protection group.
   Type: AID
   Range: SwitchAndTmgProtGrpAID or CtlProtGroupAID or EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID
   All actual existing equipment protection groups will be returned with a specific AID:
   - SwitchAndTmgProtGrpAID for high order switch fabric and timing group,
   - CtlProtGroupAID for system controller group,
   - EP155ProtGroupAID for electrical 155 MBit/s groups,
   - EP51ProtGroupAID for electrical 51 MBit/s groups,
   - LOXCProtGroupAID for low order switch fabric groups.

- TMUXProtGroupAID for TransMUX groups
- ES64UProtGroupAID for es64u groups

Default: No default

3. **condtype** – Condition Type
   This parameter specifies/identifies the type of event indication being reported. The value GP (General Purpose) is used for all 1+1 equipment protection switch events.
   Type: Alphanumeric value set.
   Range: GP.
4. **condeff** – Condition Effect
   This parameter indicates the effect of the reported event on the NE. The value TC – (Transient Condition) is used for all protection switch events.
   Type: Alphanumeric value set.
   Range: TC.
5. **ocrdat** – Occurrence Date
   This parameter indicates the date of the switch event being reported.
   Type: value set of format MM-DD.
   Range: M, D are digits. Leading zeros are not suppressed. MM represents month, DD the day.
6. **ocrtm** – Occurrence Time
   This parameter indicates the time of the switch event being reported.
   Type: value set of format HH-MM-SS.
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
7. **conddescr** – Condition Description
   This parameter provides additional information w.r.t. the protection event.

   The string value depends on the actual switch request state as follows

   - *swreq*: *condescr*
   - EQPTFAILED: \“Protection switch, EQPT, Equipment failed”\
   - FS: \“Protection switch, EQPT, Forced switch”\
   - MS: \“Protection switch, EQPT, Manual switch”\
   - NR: \“Protection switch, EQPT, No request”\

   Type: inner string.
   Range: See above. Note: For protection type EQPTCTL swreq=FS or condescr =\“Protection switch, EQPT, Forced switch\” is not applicable.
   Initial PROTNSW:
   The system evaluates the current fault information at the protection group creation time. Based on that, the system calculates and reports the switch request state. The value will be either ’NR’ or ’EQPTFAILED’ .

Command parameters (multiplex section/line protection group, REPT EVT)

1. **`modifier`** – Modifier
   The parameter is derived from the interface rate of the ports within the protection group. It shall indicate the same value as for the two ports.
   Type: Alphanumeric value set.
   Range: OC3, OC12, OC48, OC192, OC768, STM1, STM4, STM16, STM64, STM256
   Default: The default value is the interface rate of the worker port.
2. **`aid`** – Access Identifier
   This parameter specifies the port AID being a member of the protection group.
   It is derived from the active unit state as follows:
   a. current value of *actunit*, *aid* value
   b. WKG, worker port aid
   c. PROTN, protection port aid
   Type: AID
   Range: SPortAID
   Initial PROTNSW:
   The worker port AID is used in the initial protection switch notification.
3. **`condtype`** – Condition Type
   This parameter identifies the type of event indication being reported. The value GP (General Purpose) is used independent for the operation mode if the active unit state has not changed.
   If active unit state has changed and operation mode is revertive, the following values is used: The value WKSWPR-2 is used if service is switched to the protection section/line. The value WKSWBK is used if service is switched to the worker section/line.
   If active unit has changed and operation mode is non-revertive, the value PS is used.
   Type: Alphanumeric value set.
   Range: GP, WKSWPR-2, WKSWBK, PS
   Initial PROTNSW:
   The value ’GP’ is used in the initial protection switch notification.
4. **`condeff`** – Condition Effect
   This parameter indicates the effect of the reported security event on the NE. The value TC – (Transient Condition) is used for all protection switch events.
   Type: Alphanumeric value set.
   Range: TC
5. **`ocrdat`** – Occurrence Date
   This parameter indicates the date of the security event being reported.
   Type: value set of format MM-DD
   Range: M, D are digits. Leading zeros are not suppressed. MM represents month, DD the day.

6. **`ocrtm`** – Occurrence Time
   Description The parameter shall indicate the time of the security event being reported.
   Type: value set of format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
7. **`conddescr`** – Condition Description
   This parameter provides additional information w.r.t. the protection event. The string value depends on the actual switch request state as follows:
   - *swreq*: *conddescr*
   - LO: \“Protection switch, {rr} port, Lockout switch,, 1+1/1:1\”
   - FS: \“Protection switch, {rr} port, Forced switch,, 1+1/1:1\”
   - FOP: \“Protection switch, {rr} port, FOP,, 1+1/1:1\”
   - SF: \“Protection switch, {rr} port, SF,, 1+1/1:1\”
   - SD: \“Protection switch, {rr} port, SD,, 1+1/1:1\”
   - MS: \“Protection switch, {rr} port, Manual switch,, 1+1/1:1\”
   - WTR: \“Protection switch, {rr} port, WTR,, 1+1/1:1\”
   - DNR: \“Protection switch, {rr} port, DNR,, 1+1/1:1\”
   - NR: \“Protection switch, {rr} port, No request,, 1+1/1:1\”

   The value for variable *rr* is derived from the port rate within the protection group.
   Type: inner string.
   Range: See description above.
   Initial PROTNSW:
   The value ’NR’ is used for *swreq* in the initial protection switch notification.

Command parameters (2-fibre MS-SPRing/BLSR protection group, REPT EVT)

Protection switch events are reported separately for each side within the protection group. The *aid* parameter can be used in order to identify the side for which the switch event is reported.

1. **`modifier`** – Modifier
   The parameter is derived from the interface rate of the ports within the protection group. It shall indicate the same value as for the two ports.
   Type: Alphanumeric value set.
   Range: OC48, OC192, OC768, STM16, STM64, STM256
   Default: The default value is the interface rate of the worker port.
2. **`aid`** – Access Identifier
   This parameter specifies for which ring side the corresponding switch request is active; it shall either specify the east port AID or the west port AID within the protection group.
   Type: AID.
   Range: SPortAID.

3. **`condtype`** – Condition Type
   This parameter identifies the type of event indication being reported. The value WKSWPR-2 is used if service is switched to the protection route. The value WKSWBK is used if service is switched to the worker route. The value GP (General Purpose) is used if the active route has not changed.
   Type: Alphanumeric value set.
   Range: GP, WKSWPR-2, WKSWBK
   Initial PROTNSW: The value *WKSWBK* is used in the initial protection switch notification.
4. **`condeff`** – Condition Effect
   This parameter indicates the effect of the reported event on the network element. The value TC – (Transient Condition) is used for all protection switch events.
   Type: Alphanumeric value set.
   Range: TC
5. **`ocrdat`** – Occurrence Date
   The parameter shall indicate the date of the switch event being reported.
   Type: value set of format MM-DD
   Range: M, D are digits. Leading zeros are not suppressed. MM represents month, DD the day.
6. **`ocrtm`** – Occurrence Time
   The parameter shall indicate the time of the switch event being reported.
   Type: value set of format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
7. **`conddescr`** – Condition Description
   This parameter provides additional information w.r.t. the protection event. The string value depends on the actual swreq as follows:
   - *swreq*: *condescr*
   - LPS: \“Protection switch, {rr} port, Lockout switch, LPS, 2F”\
   - FSR: \“Protection switch, {rr} port, Forced switch, FSR, 2F”\
   - SFR: \“Protection switch, {rr} port, Signal fail, SFR, 2F”\
   - SDR: \“Protection switch, {rr} port, Signal degrade, SDR, 2F”\
   - MSR: \“Protection switch, {rr} port, Manual switch, MSR, 2F”\
   - WTR: \“Protection switch, {rr} port, Wait to restore, WTR, 2F”\
   - RRS: \“Protection switch, {rr} port, Reverse request, RRS, 2F”\

- RRR: \“Protection switch, {rr} port, Reverse request, RRR, 2F”\
- NR: \“Protection switch, {rr} port, No request, NR, 2F”\

The value for variable *rr* is derived from the port rate within the protection group. See value range definition for the attribute *pgrate* in the “RTRV-PROTN-GRP: Retrieve protection group” (p. 3-2759) output section w.r.t. supported port rates for the different protection types.
Type: inner string
Range: See description above.
Initial PROTNSW: The value ’NR’ is used for *swreq* in the initial protection switch notification.

Command parameters (4-fibre MS-SPRing/BLSR protection group, REPT EVT)

Protection switch events are reported separately for each side within the protection group. The aid parameter can be used in order to identify the side for which the switch event is reported.

1. **`modifier`** – Modifier
   The parameter is derived from the interface rate of the ports within the protection group. It shall indicate the same value as for the two ports.
   Type: Alphanumeric value set.
   Range: OC48, STM16, OC192, STM64. If *protnmode* = TOPEX, then the range is restricted to: STM64 only
   Default: The default value is the interface rate of the worker port.
2. **`aid`** – Access Identifier
   This parameter specifies the port AID being a member of the protection group.
   It is derived from the active unit state for each side (east or west) as follows:
   a. current value of *actunit*, *aid* value
   b. WKG, worker port aid
   c. PROTN, protection port aid
      The attributes switch request state and active unite state may be updated in the context of a switch event. Actual means the value of the parameter after the switch has occurred.

   Type: AID.
   Range: SPortAID.
   Initial PROTNSW: The worker port AID is used in the initial protection switch notification.
   Asymmetric Ring Support: In all protection switch notifications for an asymmetric span, where a worker port AID is required, but the worker port is not available, the AID of the protection port of the same side of the protection group is used.
3. **`condtype`** – Condition Type
   This parameter identifies the type of event indication being reported. The value WKSWPR-ep is used if service is switched to east protection span.
   The value WKSWBK-wp is used if service is switched to west protection span.
   The value WKSWBK is used if service is switched to working. The value GP

(General Purpose) is used if the active route/span has not changed.
Type: Alphanumeric value set.
Range: GP, WKSWPR-ep, WKSWPR-wp, WKSWBK
Initial PROTNSW: The value *WKSWBK* is used in the initial protection switch notification.

4. **condeff** – Condition Effect
he parameter shall indicate the effect of the reported event on the network element. The value TC – (Transient Condition) is used for all protection switch events.
Type: Alphanumeric value set.
Range: TC
5. **ocrdat** – Occurrence Date
The parameter shall indicate the date of the switch event being reported.
Type: value set of format MM-DD
Range: M, D are digits. Leading zeros are not suppressed. MM represents month, DD the day.
6. **ocrtm** – Occurrence Time
The parameter shall indicate the time of the switch event being reported.
Type: value set of format HH-MM-SS
Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
7. **conddescr** – Condition Description
This parameter provides additional information w.r.t. the protection event. The string value depends on the actual swreq as follows:
   - *swreq*: *condescr*
   - LPS: \“Protection switch, {rr} port, Lockout switch, LPS, 4F”\
   - FSS: \“Protection switch, {rr} port, Forced switch, FSS, 4F”\
   - FSR: \“Protection switch, {rr} port, Forced switch, FSR, 4F”\
   - SFS: \“Protection switch, {rr} port, Signal fail, SFS, 4F”\
   - SFR: \“Protection switch, {rr} port, Signal fail, SFR, 4F”\
   - SDS: \“Protection switch, {rr} port, Signal degrade SDS, 4F”\
   - SDR: \“Protection switch, {rr} port, Signal degrade, SDR, 4F”\
   - MSS: \“Protection switch, {rr} port, Manual switch, MSS, 4F”\
   - MSR: \“Protection switch, {rr} port, Manual switch, MSR, 4F”\
   - WTR: \“Protection switch, {rr} port, Wait to restore, WTR, 4F”\
   - RRS: \“Protection switch, {rr} port, Reverse request, RRS, 4F”\

- RRR: \“Protection switch, {rr} port, Reverse request, RRR, 4F”\
- NR: \“Protection switch, {rr} port, No request, NR, 4F”\

The value for variable *rr* is derived from the port rate within the protection group. See value range definition for the attribute *pgrate* in the “RTRV-PROTN-GRP: Retrieve protection group” (p. 3-2759) output section w.r.t. supported port rates for the different protection types.
Type: inner string.
Range: See Description.
Initial PROTNSW: The value ’NR’ is used for *swreq* in the initial protection switch notification.

Command parameters (Path protection group, REPT EVT)

1. **modifier** – Modifier
The parameter is derived from the interface rate of the ports within the protection group. It shall indicate the same value as for the two ports.
Type: Alphanumeric value set.
Range: STS1, STS3C, STS12C, STS48C, STS192C, VC3, VC4, VC44C, VC416C, VC464C
Default: The default value is derived from the tributary input signal rate list in case of fixed mode.
2. **aid** – Access Identifier
This parameter specifies for the output tributary AID of the constituent path selector AID being a member of the path protection group.
Note: If path protection group consists only of one constituent path selector, the output tributary AID is equal to the path protection group AID.
Type: AID.
Range: STS1NumAID or VCGNumAID or GenLOTribAID.
The use of wild cards is not allowed for any part of the AID.
3. **condtype** – Condition Type
This parameter specifies the type of event indication being reported. The value GP (General Purpose) is used if the active unit state has not changed.
If active unit state has changed, the following values are used: The value WKSWPR-2 is used if service is switched to the protection tributary. The value WKSWBK is used if service is switched to the worker tributary.
Type: Alphanumeric value set.
Range: GP, WKSWPR-2, WKSWBK.
Initial PROTNSW: The value ’GP’ is used in the initial protection switch notification.
4. **condeff** – Condition Effect
The parameter shall indicate the effect of the reported event on the network element. The value TC – (Transient Condition) is used for all protection switch events.
Type: Alphanumeric value set.
Range: TC

5. **ocrdat** – Occurrence Date
   The parameter shall indicate the date of the switch event being reported.
   Type: value set of format MM-DD
   Range: M, D are digits. Leading zeros are not suppressed. MM represents month, DD the day.
6. **ocrtm** – Occurrence Time
   The parameter shall indicate the time of the switch event being reported.
   Type: value set of format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
7. **conddescr** – Condition Description
   This parameter provides additional information w.r.t. the protection event. The string value depends on the actual switch request state as follows:
   - *swreq*: *conddescr*
   - LO: \“Protection switch, {rr} CS, Lockout switch”\
   - FS: \“Protection switch, {rr} CS, Forced switch”\
   - SF: \“Protection switch, {rr} CS, SF”\
   - SD: \“Protection switch, {rr} CS, SD”\
   - MS: \“Protection switch, {rr} CS, Manual switch”\
   - WTR: \“Protection switch, {rr} CS, WTR”\
   - NR: \“Protection switch, {rr} CS, No request”\

   The value for variable *rr* is derived from the constituent signal type; i.e. it has the following values: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C.
   Type: inner string.
   Range: See description above.
   Initial PROTNSW: The value ’NR’ is used for *swreq* in the initial protection switch notification.

Example output

1+1 Equipment protection group

he following example reports a manual switch to protection circuit pack for 1+1 equipment protection group. Because the manual switch chnges the active unite state and the switch request state, a REPT SW Notification is generated in addition to the REPT EVT Notification:

```
   LT-UNITE 99-01-10 08:12:00
A  001 REPT SW
   ″1-1-#-#-xcp-cp,1-1-#-#-xcw-cp″
;

   LT-UNITE 98-01-10 08:12:00
A  002 REPT EVT COM
```

```
   "1-1-estgrp:GP,TC,01-10,08-12-00:\"Protection Switch, EQPT, Manual switch\""
;
```

Multiplex section/line protection group (1+1 and 1:1)

The following example reports a lockout switch for a 1+1 multiplex section/line protection group:

```
   LT-UNITE 98-01-10 08:12:00
A  015 REPT EVT OC3
   "1-1-#-#-10-2:WKSWBK,TC,01-10,08-12-00:\"Protection Switch, OC3 port, Lockout
      switch,, 1+1/1x1\""
;
```

2 fibre MS-SPRing/BLSR protection group

The following example reports a manual ring switch of the east side for a 2F MS-SPRing/BLSR protection group:

```
   LT-UNITE 98-01-10 08:12:00
A  015 REPT EVT OC48
   "1-1-#-#-10-2:WKSWPR-2,TC,01-10,08-12-00:\"Protection Switch, OC48 port, Manual
      switch, MSR, 2F\""
;
```

4 fibre MS-SPRing/BLSR protection group

The following example reports a manual span switch of the east side for a 4F MS-SPRing/BLSR protection group:

```
   LT-UNITE 98-01-10 08:12:00
A  015 REPT EVT OC192
   "1-1-#-#-10-1:WKSWPR-2,TC,01-10,08-12-00:\"Protection Switch, OC192 port, Manual
      switch, MSS, 4F\""
;
```

Path protection group

The following example reports a autonomous switch to protection for a 1+1 path protection group:

```
   LT-UNITE 98-01-10 08:12:00
A  015 REPT EVT STS1
   "1-1-#-#-10-1-25:WKSWPR-2,TC,01-10,08-12-00:\"Protection Switch, STS1 CS, SF\""
;
```

REPT ALM: Report alarm

REPT ALM messages are generated autonomously by the NE to report an alarm that requires immediate attention by the craft at the OS. Each alarm message has an alarm severity associated with it.

Output format

The general output format of the **REPT ALM** message is defined as follows:

```
sid date time
almcde atag REPT ALM modifier
  ″aid:ntfcncde,condtype,srveff,ocrdat,ocrtm:conddescr″
   (1 or more lines of the above)
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the output parameter section for the **RTRV-HDR** command.

Output parameters

Command parameters

1. **almcde** – Alarm Code
   The alarm code indicates the severity of the reported alarm or if the alarm is cleared.
   Type: Value set.
   Range:
   - *C: Critical alarm
   - **: Major alarm
   - *: Minor alarm
   - A: Automatic message (nonalarm). This is used only to report a cleared alarm.
2. **atag** – Automatic Tag
   This parameter is used for message sequencing. The number is incremented by one for each autonomous message sent by the NE.
   Type: Integer (3 digits).
   Range: 000 – 999 Leading zeros are not suppressed.
3. **modifier** – Message Modifier
   This parameter defines the AID type to which the alarm belongs.
   Type: Alphanumeric value set.
   Range:
   - COM: Common. An alarm/event that applies to the whole NE system; for example, a processing error.
   - EQPT: An equipment-related alarm/event.
   - SYST: An alarm/event related to system timing.

- OUTT: An alarm/event related to output timing.
- STM1E: A facility-related alarm/event for synchronous transport mode 1 (electrical).
- STM1: A facility-related alarm/event for synchronous transport mode 1.
- STM4: A facility-related alarm/event for synchronous transport mode 4.
- STM16: A facility-related alarm/event for synchronous transport mode 16.
- STM16T: A facility-related alarm/event for fully transparent synchronous transport mode 16.
- STM64: A facility-related alarm/event for synchronous transport mode 64.
- STM256: A facility-related alarm/event for synchronous transport mode 256.
- ODU1: A facility-related alarm/event for optical data units of level 1
- OC3: A facility-related alarm/event at the optical carrier signal level 3.
- OC12: A facility-related alarm/event at the optical carrier signal level 12.
- OC48: A facility-related alarm/event at the optical carrier signal level 48.
- OC48T: A facility-related alarm/event at the fully transparent optical carrier signal level 48.
- OC192: A facility-related alarm/event at the optical carrier signal level 192
- OC768: A facility-related alarm/event at the optical carrier signal level 768
- T3: A facility-related alarm/event at the DS3 level
- TMAT3: A facility-related alarm/event at the DS3 level related to TransMUX function
- EC1: A facility-related alarm/event at the EC1 level
- 1GE: A facility-related alarm/event for 1 Gigabit Ethernet.
- 1GEPL: A facility-related alarm/event for 1 Gigabit Ethernet Private Line.
- 10GEPL: A facility-related alarm/event for 10 Gigabit Ethernet Private Line.
- VC3: A facility-related alarm/event for virtual container 3.
- VCG: A facility-related alarm/event for virtual concatenation group.
- VC4: A facility-related alarm/event for virtual container 4
- VC44C: A facility-related alarm/event for a signal concatenated at 4 times the VC4 rate.
- VC416C: A facility-related alarm/event for a signal concatenated at 16 times the VC4 rate.
- VC464C: A facility-related alarm/event for a signal concatenated at 64 times the VC4 rate.
- STS1: A facility-related alarm/event at synchronous transport signal level 1.
- TMASTS1: A facility-related alarm/event at synchronous transport signal level 1 related to TransMUX function.

- STS3C: A facility-related alarm/event at synchronous transport signal level 3.
- STS12C: A facility-related alarm/event at synchronous transport signal level 12.
- STS48C: A facility-related alarm/event at synchronous transport signal level 48.
- STS192C: A facility-related alarm/event at synchronous transport signal level 192.
- HOVC4 : A facility-related alarm/event for virtual container 4 higher-order path termination.
- HOSTS1: A facility-related alarm/event at synchronous transport signal level 1 higher-order path termination.
- LOVC3: A facility-related alarm/event for virtual container 3 (lower-order path).
- LOVC12: A facility-related alarm/event for virtual container 12
- LOVT1: A facility-related alarm/event at transport level VT1.5.
- TMSVT: A facility-related alarm/event at transport level VT1.5 related to TransMUX function.
- DS1: A facility-related alarm/event at transport level DS1.
- LINK: A signaling communication link related alarm/event for ONNS

Default: None.

4. **`aid`** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: All defined AIDs are allowed. It should be a valid AID for the specified *modifier*. Refer to the definition of AIDs.
5. **`ntfcncde`** – Notification Code
   This is the alarm severity of the reported Alarm or the clear information of a cleared Alarm.
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)
   - MN – Minor (ITU / Telcordia)
   - CL – Cleared.
6. **`condtype`** – Condition Type
   This parameter identifies the probable cause (type of alarm indication) being reported.
   Type: Alphanumeric value set.
   Range: All defined probable causes.
   `srveff` – Service Effect
   This parameter identifies the service affect of alarms being reported. It presents the “best guess” from a local system point of view.
   Type: Alphanumeric value set.

Range:

- SA: Service-Affecting alarm condition
- NSA: Non-Service-Affecting alarm condition.

7. **`ocrdat`** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
8. **`ocrtm`** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
9. **`conddescr`** – Description of the Condition
   This is a textual Description of the Condition.
   Type: inner string of the format:
   \″<category>, <entity>, <description>, <additional text>\″ or
   \″<category>, <entity>, <description>, <additional text>-cleared\″ (this form is used in REPT ALM/EVT for cleared alarms)
   Range:
   - category: is the category
   - entity: is the entity for which the alarm is reported
   - description: is the Alarm description, the string is up to 32 characters
   - additional text: additional text.

   If there is no additional text, then the trailing comma may be omitted.

Detailed behavior description

If an alarm has an severity of critical, major, or minor, then its occurrence will be reported in a REPT ALM message. An exception to this rule is an environmental alarm which is reported by a REPT ALM ENV autonomous message.

An alarm reported to the OS via the REPT ALM message has a corresponding REPT ALM clearance message that is generated when the alarm clears.

Example input/output

The following example shows an autonomous message REPT ALM for an LOS Alarm:

```
   LT-UNITE-123 00-01-01 08:00:00
*C 126 REPT ALM OC48
  ″1-1-#-#-3-1:CR,LOS,SA,01-01,07-29-13:\″Communication-Transport, OC48 port, Loss
      Of Signal\″″
```

```
;
```

The following example shows an autonomous message REPT ALM for an LOS Alarm clear:

```
   LT-UNITE-123 00-01-01 08:00:00
A  127 REPT ALM OC48
  ″1-1-#-#-3-1:CL,LOS,SA,01-01,07-29-13:\″Communication-Transport, OC48 port, Loss
      Of Signal-cleared\″″
;
```

Related TL1 messages

- "ALW-MSG: Allow message" (p. 3-33)
- "ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
- "INH-MSG: Inhibit message" (p. 3-2021)
- "REPT-ALM-ENV: Report alarm environment" (p. 3-2101)
- "REPT EVT: Report event" (p. 3-2121)
- "RTRV-ALM: Retrieve alarm" (p. 3-2231)
- "RTRV-ALM-ENV: Retrieve alarm environment" (p. 3-2238)
- "RTRV-COND: Retrieve condition" (p. 3-2285)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: NE and Transport Surveillance Messages

REPT-ALM-ENV: Report alarm environment

The **REPT ALM ENV** autonomous messages report when environmental alarm of severity CR (critical), MJ (major), or MN (minor) either sets or clears.

Output format

The general output format of the **REPT ALM ENV** message is defined as follows:

```
   sid date time
almcde atag REPT ALM ENV
  "aid:ntfcncde,almtype,ocrdat,ocrtm[,almmsg]"
   (1 or more lines of the above)
;
```

Output parameters

Command parameters

1. **almcde** – Alarm Code
   The alarm code indicates the severity of the reported alarm or if the alarm is cleared.
   Type: Value set.
   Range:
   - *C: Critical alarm
   - **: Major alarm
   - *: Minor alarm
   - A: Automatic message (nonalarm). This is used only to report a cleared alarm.
2. **atag** – Automatic Tag
   This parameter is used for message sequencing. The number is incremented by one for each autonomous message sent by the NE.
   Type: Integer (3 digits).
   Range: 000 – 999 Leading zeros are not suppressed.
3. **aid** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: Miscellaneous_in_discreteAID. The AID misc_inall must not appear.
4. **almtype** – Alarm Type
   This parameter identifies the type of alarm indication being reported.
   Type: Alphanumeric value set.
   Range: MISC (default), an event that applies to the miscellaneous discrete outputs inputs.

5. **`ocrdat`** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
6. **`ocrtm`** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
7. **`almmsg`** – Alarm Message
   A string describing the alarm of the environmental point.
   Note: This string is configurable by the **`SET-ATTR-ENV`** command.
   Type: text string.
   Range: upper and lower case, up to 32 characters.
   Default: Miscellaneous Discrete.

Detailed behavior description

REPT ALM ENV messages will be reported if on environmental points (MDI's) an alarm of severity CR (critical), MJ (major), or MN (minor) will be raised.

An alarm reported to the OS via the REPT ALM ENV message has a corresponding REPT ALM ENV clearance message that is generated when the alarm condition clears.

An environmental alarm of severity NA (warning (ITU) / not alarmed (telcordia) will be reported by REPT EVT.

Example input/output

The following example shows a REPT ALM ENV message if an environmental alarm appears:

```
   LT-UNITE-123 00-01-01 08:00:00
*C 126 REPT ALM ENV
  ″MISC_IN7:CR,MISC,01-01,07-29-13,\″open door\″″
;
```

The following example shows a REPT ALM ENV message if an environmental alarm disappears:

```
   LT-UNITE-123 00-01-01 08:00:00
A  127 REPT ALM ENV
  ″MISC_IN7:CL,MISC,01-01,07-29-13,\″open door\″″
;
```

Related TL1 messages

- “ALW-MSG: Allow message” (p. 3-33)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “INH-MSG: Inhibit message” (p. 3-2021)
- “REPT ALM: Report alarm” (p. 3-2096)
- “REPT EVT: Report event” (p. 3-2121)
- “REPT SW: Report switch” (p. 3-2188)
- “RTRV-ALM: Retrieve alarm” (p. 3-2231)
- “RTRV-ALM-ENV: Retrieve alarm environment” (p. 3-2238)
- “RTRV-COND: Retrieve condition” (p. 3-2285)
- “SET-ATTR-ENV: Set attribute environment” (p. 3-3137)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

REPT CONNCHG notification

Overview

The **REPT CONNCHG** notification is used to notify the setup or tear down of Network Connections via multiple domains triggered via UNI. UNI triggered setup/tear down of single domain connections is not notified.

The REPT CONNCHG notification is also used to notify about TL1 triggered changes (ENT/ED/DLT-CONN) or autonomously triggered changes of Network Connection parameters.

The REPT CONNCHG notification message reports specific changes in the system data repositories of the ingress node of every involved domain. Changes can be caused by a user's TL1 command, UNI request or by autonomous internally generated connection changes.

Notification messages resulting from a TL1 command or UNI request

If the data repository change occurs as a result of a TL1 ENT-CONN, DLT-CONN, ED-CONN command or a UNI request, then a REPT CONNCHG message is transmitted to all users provisioned to receive REPT CONNCHG messages. Therefore, a management OS must expect to receive data repository change messages for changes it caused, caused by a CIT user as well as those caused by a UNI request. The REPT CONNCHG CCB field will indicate the TL1 command that caused the connection change.

Autonomous Network Connection configuration change REPT CONNCHG format

Autonomous network connection configuration changes that affect the data repository will cause a REPT CONNCHG messages that are output with the CCB = CONN.

Output format

Output format for a REPT CONNCHG notification message

The general format of a REPT CONNCHG notification is as follows:

```
   srcTid date time
A  atag REPT CONNCHG
   ″TIME=ocrTime, DATE=ocrDate:CCB=ccb:srcASTNNodeID:msgCode,eventText″
   ″[CallID=callID],[ConnID=connID]:specBlock″;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in RTRV-HDR.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. **`date`**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date

3. **`time`**
   Time. The time parameter is set to the time when the output message is generated. See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time

4. **`atag`** – Automatic Tag
   The *atag* is used for message sequencing. The *atag* is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Type: decimal number
   Range: 000-999
   Default: 0

5. **`ocrDate`** – Occurence Date
   Date. The date parameter is set to the date when the event occurred in the system.
   Type: Value set.
   Range:

YY-MM-DD, where

- YY=00-99,
- MM=01-12,
- DD=01-31

Default: occurence date

6. **`ocrTime`** – Occurence Time

   Time. The time parameter is set to the time when the event occurred in the system.
   Type: Value set.
   Range:
   HH:MM:SS, where

   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: occurrence time

7. **`comBlock(CB)`**- CCB common block

   This block contains the parts of the command (TL1 or UNI request) causing this REPT CONNCHG notification plus further event info. The following parameters, separated by : will be present:

   - CCB – command code block for the command which caused the event
   - srcASTNNodeID – srcASTNNodeID input parameter to the command
   - msgCode – msgCode of the event
   - eventText – further event info.

   a. **`CCB`** – Command Code Block

      Represents the network connection command (TL1 or UNI request) causing the change. This confirmation notification indicates a network connection change was stored in the ONNS Host NVM.

      Type: Value set.

      Range:

      CONN is specified when the change is not directly caused by a network connection command (TL1 or UNI request), e.g. forced deletion other values are used when the change is directly caused by a user TL1 or UNI request command:
      – ENT-CONN
      – ED-CONN
      – DLT-CONN

   b. **`srcASTNNodeID`**

      ONNS source nodeID. Together with the following *srcPortID* field defines the associated Ingress *connPoint* that connects the source node to the edge/gateway switch or client router.

      Type: Four dot separated numeric strings

      Range: for each numeric string: 0..255

c. **`msgCode`** – Message Code

Message Code or Status code indicates known or suspected cause or reason for the event, change, or condition.

Type: Alphanumeric string of length 4

Range: See for possible message codes in chapter “CONNCHG events”.

d. **`eventText`**

This field records an event description message.

Type: text string

Range: See for possible message codes in chapter “CONNCHG events”.

e. **`CONNCHG events`**

EVENTS SPECIFIC TO ENT-CONN:

Successful Network Connection setup

Event: C001 – Network Connection successfully setup

Calling Party Busy

Event: E001 – Ingress address already in use

Called Party Busy

Event: E002 – Egress address already in use

Unavailable SNP

Event: E004 – Selected timeslot on eNNI link already in use

Error code “Unavailable SNP” shall include error code “Invalid SNP”, as long as internally both are not distinguishable, because “Unavailable SNP” is considered as the more generic one.

Unavailable SNPP

Event: E006 – Selected port on eNNI link already in use

Error code “Unavailable SNPP” shall include error code “Invalid SNPP”, as long as internally both are not distinguishable, because “Unavailable SNPP” is considered as the more generic one.

Invalid SPC SNP

Event: E007 – Invalid Egress address

The event “Invalid SPC SNP” shall especially be applied in case input parameters *destTNAAddress* does not address a network TNA address information (that is, does not address a TNA information for the SDH/SONET layer).

Invalid route

Event: E008 – Invalid Level1 Routing decision

Service affecting defect

Event: E009 – Service affecting defect

Non service affecting defect

Event: E010 – Non service affecting defect

Unavailable service level

Event: E011 – Service level is unavailable

Unavailable directionality

Event: E012 – Directionality is unavailable

Unauthorized sender (policy error)

Event: E013 – Unauthorized sender (policy error)

Unauthorized receiver (policy error)

Event: E014 – Unauthorized receiver (policy error)

RSVP protocol failure

The output format is shown below, where **`<error code>`** is a 2-character code and **`<value>`** is a 3-character code describing the RSVP defined error codes and values.

Event: E015 – Protocol failure, <error code> / <value>

Input Parameter Source Label not valid

The following event is raised if the value for the input parameter *srcLabel* is not valid.

Event: E016 – invalid Source Label

Input Parameter Destination TNA Address not valid

The following event is raised if the value for the input parameter *destTNAAddress* is not valid.

Event: E017 – invalid Destination TNA Address

EVENTS SPECIFIC TO DLT-CONN:

Graceful deletion of a Network Connection

Event: C060 – Network Connection graceful deleted

EVENTS INDEPENDENT TO NETWORK CONNECTION COMMANDS

Forced Deletion of a Network Connection

Event: C100 – Network Connection forced deleted

8. **`callID`**
   The *callID* indicates a call of a logical call / connection separation (call capability). This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 .. F).

9. **`connID`**
   Connection ID for an existing ONNS connection segment of a UNI or Network connection. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.

Spec Block Parameters

The following spec block parameters are only visible in notifications created as result of an ENT-CONN or eNNI setup request.

1. **`bandwidth`**
   Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
   Type: Value set.
   Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
   Default: None.

   This is an output parameter in **`REPT CONNCHG`** in the following cases:

   - CCB = **`ENT-CONN`**

2. **`destLabel`**
   The destination label defines together with the destination logical port ID the associated egress *connPoint* that connects the destination node of the Network Connection. The *destLabel* is defined in KLM notation. *destLabel* is the requested setup parameter (via ENT-CONN or UNI) and is only visible at the source node of the Network Connection.
   Type: Character string.
   Range:

   Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:

   - S #0..65535
   - U #0..15
   - K #0..15
   - L #0..15
   - M #0..15

   Example for an STS-12c connection service on a OC48 link which starts at timeslot 25: 9-0-0-0-0.
   Destination Label 65535-15-15-15-15 is reserved for GbE connections.
   destLabel is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

3. **`destLogicalPortID`**
   destLogicalPortID is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

4. **`destIPv4TNAAddr`**
   Identifier of a TNA address in IPv4 format.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0 .. 255
   destIPv4TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is IPv4TNAADDR.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

5. **`destIPv6TNAAddr`**
   Identifier of a TNA address in IPv6 format.
   Type: Eight hyphen separated hexadecimal numbers
   Range: for each hexadecimal number: 0 .. FFFF
   destIPv6TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is IPv6TNAADDR.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

6. **`destNSAPTNAAddr`**
   Identifier of a TNA address in NSAP format.
   Type: Alphanumeric value set (20 byte)
   Range: up to 20 alphanumeric strings with hexadecimal values 00..FF
   destNSAPTNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is NSAPTNAADDR.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

7. **`dir`**
   Path Direction Topology:

   - Unidirectional A-Z (1WAY) or
   - Bidirectional (2WAY)

   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default) or 2WAY

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

8. **`routingList`**

   This parameter describes the level 1 routing described as a list of level 1 nodeIDs and linkIDs. The *routingList* retrieved from an intermediate or destination domain will not show the *localAdvNodeID – localLinkID* combination of preceding domains.
   Type: Character string.
   Range:

   hyphen separated list of numeric string pairs, where the first numeric string represents a *localAdvNodeID* and the second a *localLinkID* with:

   - *localAdvNodeID*: four dot separated numeric string, each numeric string with the values 0..255
   - *localLinkID*: numeric string, with the values 0..42946967295

   at maximum 10 *localAdvNodeIDs* and 10 *localLinkIDs*.
   The hyphen is used as separator between the two numeric strings as also between the string pairs.
   Example for a *routingList* describing level 1 Routing via 3 domains at the source domain: 126.43.231.4-643-1.232.3.1-4209-1.4.23.8-5.
   routingList is an optional parameter and is only visible for Network Connections via multiple domains.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

9. **`exclLinkResCls`**

   The exclude list of *linkResCls* is used to exclude interdomain links from the interdomain routing calculation. *exclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
   Type: Alphanumeric value set (4 bytes), every bit defining one link class
   Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
   exclLinkResCls is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.

   This are an output parameter in REPT CONNCHG in the following cases:

   - CCB=ENT-CONN

10. **`genLabelRequest`**

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

11. **`inclLinkResCls`**
    The include list of *linkResCls* is used to define interdomain links which has to be included for interdomain routing calculation. *inclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    inclLinkResCls is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

12. **`ingressLSP`**
    Local Connection ID for the connection segment of a Network Connection between source UNI-C to UNI-N or eNNI-U to eNNI-D.
    srcLSP is formed from the Tunnel_ID of the UNI_IPv4_SESSION/eNNI_IPv4_SESSION object and the LSP_ID of the LSP_TUNNEL_IPv4_SENDER_TEMPLATE of this connection segment.
    Type: character string of length up to 1
    Range:

    TunnelID-LSPID, with

    - TunnelID: numeric string: 0..65535
    - LSPID: numeric string: 0..65535

    Example: 2319-321

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

13. **`policyData`**
    Policy data may in general include credentials identifying users or user classes, account numbers, limits, quotas, etc. to support access policies.
    Type: Alphanumeric value set (60 bytes)
    Range: up to 60 alphanumeric characters with hexadecimal values 00..FF

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

14. **`protType`**
    This parameter defines the requested protection of advertised abstract links (interdomain as well as intradomain).
    It is the requested setup parameter (via **`ENT-CONN`**) and is only visible at the source node if requested.
    Type: Alphanumeric value set.

    Range:

    - 0x1: only unprotected links shall be considered for routing
    - 1+1: only 1+1 MSP protected links shall be considered for routing)
    - N/A: all links shall be considered for routing, irr. of their protection type

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

15. **`remoteTID`**
    This parameter provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the RTRV-HDR description.
    remoteTID is an optional parameter and is not visible for Network Connections running in a single domain. It shall also not be visible in the destination domain for Network Connections running via multiple domains.

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

16. **`servLevel`**
    The service level is mapping the connection type of an ONNS connection segment of the Network Connection. The values of the connection type are representing a mapping of the ONNS parameters *connectionClass*, *disjointness* and *grpSetupType* (for Ethernet connections).
    Type: Numeric string.
    Range: 0..255

    This are an output parameter in REPT CONNCHG in the following cases:

    - CCB=ENT-CONN

17. **`srcLabel`**
    The source label defines together with the source logical port ID the associated ingress *connPoint* that connects the source node of the Network Connection. The *srcLabel* is defined in KLM notation.
    srcLabel is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
    Type: Character string.
    Range:

    Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:

    - S #0..65535
    - U #0..15
    - K #0..15

- L #0..15
- M #0..15

Example for an STS-12c connection service on a OC48 link which starts at timeslot 25: 9-0-0-0-0.
Source Label 65535-15-15-15-15 is reserved for GbE connections.
srcLabel is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

18. **`srcLogicalPortID`**
Logical Port Identifier related to the TNA address. *srcLogicalPortID* is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
Type: numeric value set
Range: 0 to 4294967295

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

19. **`scrIPv4TNAAddr`**
Identifier of a TNA address in IPv4 format.
Type: Four dot separated numeric strings
Range: for each numeric string: 0 .. 255
srcIPv4TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is IPv4TNAADDR.

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

20. **`srcIPv6TNAAddr`**
Identifier of a TNA address in IPv6 format.
Type: Eight hyphen separated hexadecimal numbers
Range: for each hexadecimal number: 0 .. FFFF
srcIPv6TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is IPv6TNAADDR.

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

21. **`srcNSAPTNAAddr`**
Identifier of a TNA address in NSAP format.
Type: Alphanumeric value set (20 byte)
Range: up to 20 alphanumeric strings with hexadecimal values 00..FF
srcNSAPTNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is NSAPTNAADDR.

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

22. **`tspec`** – Traffic Parameters

The range of the traffic parameters depends on the used LSP Encoding Type. The *tspec* defines information about the traffic of the connection segments between source UNI-C to UNI-N and destination UNI-N to UNI-C or between domains (eNNI-U and eNNI-D).
Type: Character string.
Range for LSP Encoding Type "SONET/SDH":

SONET/SDH tspec: five dash separated numeric strings representing SignalType#, RCC#, NCC#, NVC#, MT# with:

- SignalType #1..12
- RCC #0,1
- NCC #0..255
- NVC #0..63
- MT #1..64

Example for an STS-12c: 6-1-4-0-1-0.
SONET/SDH Tspec parameters "T" and "Profile" is not stored in NVM.
Range for LSP Encoding Type "ETHERNET":

ETHERNET tspec: five dash separated numeric strings representing Token Bucket Rate#, Token Bucket Size#, Peak Data Rate#, Minimum Policed Unit#, Maximum Packet Size# with:

- Token Bucket Rate #32-bit IEEE floating point number
- Token Bucket Size #32-bit IEEE floating point number
- Peak Data Rate #32-bit IEEE floating point number
  (e.g. 0x4CEE6B28 for 1GE)
- Minimum Policed Unit #1..FFFFFFFF
- Maximum Packet Size #1..FFFFFFFF

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

23. **`tunnelIfID`**

All SDH/SONET connections belonging to the same VCAT group are identified by the LSP Tunnel Interface ID.
Type: character string of length up to 24
Range:

LSRRouterID&InterfaceID - with:

- *LSRRouterID*: four dot separated numeric strings: each string 0..255
- *InterfaceID*: Numeric string 0..4294967295

Example: 117.32.234.13&32439

This are an output parameter in REPT CONNCHG in the following cases:

- CCB=ENT-CONN

Example output

Output format for a REPT CONNCHG notification message after ENT-CONN

The format of a REPT CONNCHG notification in the source node of the Network Connection is as follows:

```
    LT-ONNS101 2002-08-01 08:00:00
A  01135 REPT CONNCHG
   ″TIME=07-59-59,DATE=2002-08-01:CCB=ENT-CONN:125.114.48.12:C001:\″Network
      Connection successfully setup\:″
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.196.203&1-6-5-1&1:ServiceLevel=0&
Direction=2WAY&Bandwidth=STS12c&ProtType=0x1&IngressLSP=2319203&SRCIPV4TNAADDR=
135.13.3.119&SrcLogicalPortID=1019&SrcLabel=9-0-0-0-
      0&DESTIPV4TNAADDR=135.225.43.128&
DestLogicalPortID=2&DestLabel=4-0-0-0-0&RoutingList=126.43.231.4-13.121.39.21-
      1.232.3.1-12.38.217.4-1.4.23.8&GenLabelRequest=5-100-34&Tspec=6-1-4-0-1-0″;
```

Output format for a REPT CONNCHG notification message after ED-CONN

The format of a REPT CONNCHG notification in the source node of the Network Connection is as follows:

```
  LT-ONNS101 2002-08-01 08:00:00
A  01135 REPT CONNCHG
   ″TIME=07-59-59,DATE=2002-08-01:CCB=ED-CONN:125.114.48.12:C050:\″Network
      Connection Alias changed\:″
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.196.203&1-6-5-1&1,NetwConn01:″
```

Output format for a REPT CONNCHG notification message after DLT-CONN

The general format of a REPT CONNCHG notification in the source node of the Network Connection is as follows:

```
  LT-ONNS101 2002-08-01 08:00:00
A  01135 REPT CONNCHG
   ″TIME=07-59-59,DATE=2002-08-01:CCB=DLT-CONN:125.114.48.12:C060:\″Network
      Connection graceful deleted\:″
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.196.203&1-6-5-1&1,:″;
```

Output format for a REPT CONNCHG notification not directly caused by a network connection command

The general format of a REPT CONNCHG notification is as follows:

```
    LT-ONNS101 2002-08-01 08:00:00
A  01135 REPT CONNCHG
   ″TIME=07-59-59,DATE=2002-08-01:CCB=CONN:125.114.48.12:C100:\″Network Connection
      forced deleted\:″
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.196.203&1-6-5-1&1,:ServiceLevel=0&
```

```
Direction=2WAY&Bandwidth=STS12c&ProtType=0x1&IngressLSP=2319-203&SRCIPV4TNAADDR=
135.13.3.119&SrcLogicalPortID=1019&SrcLabel=9-0-0-0-0&DESTIPV4TNAADDR=
135.225.43.128&DestLogicalPortID=2&DestLabel=4-0-0-0-0&RoutingList=
126.43.231.4-13.121.39.21-1.232.3.1-12.38.217.4-1.4.23.8&GenLabelRequest=
5-100-34&Tspec=6-1-4-0-1-0";
```

Related TL1 messages

- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
- "DLT-CONN: Delete network connection" (p. 3-139)

REPT DBCHG: Report data base change

The **REPT DBCHG** message reports any change in the system database.

Output format

The **REPT DBCHG** message uses the following format:

```
   sid date time
A  atag REPT DBCHG
   "TIME=chgtm,DATE=chgdat:ccb:[aid]:[com_block]:[spec_block]"
     ...
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

1. **atag** – Automatic Tag
   The *atag* is used for message sequencing. The *atag* is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Type: decimal number.
   Range: 0 to 999.
   Default: 0
2. **chgtm** – Change Time
   Time when the change is occurred.
   Type: value set.
   Range: HH-MM-SS.
3. **chgdat** – Change Date
   Date when the change occurred
   Type: Value set.
   Range: YY-MM-DD
4. **ccb** – Command Code Block
   This field will be set to the TL1 command that causes that database change. The *com_block* and *spec_block* fields will be set according to the parameters that changed.
   Type: Alphanumeric value set verb-modifier[-modifier].
5. **aid** – Access Identifier
   AID of the entity for which the change is reported
   Type: AID.
   Range: As defined by the *aid* input parameter of the corresponding command, defined by *ccb*.

6. **`com_block`** – Common Block
   This is a position-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter values are included in this block in the same order as the corresponding TL1 command.
   As exception to this rule are some commands in the protection fragment, where the aliases *rid* and *ppgname* are not reported in the **`REPT DBCHG`** notification.
   If the database change is the result of a TL1 command, then the parameter *ctag* from the TL1 command is not included in the *com_block*.
   Type: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
7. **`spec_block`** – Specific Block
   This is a name-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter names and values are included in this block. parameters within the specific block are positionally independent and are specified using a name-defined construct of *parameter* = value in a comma-separated list. The parameter names are the same as those for the corresponding TL1 command.
   Type: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.

Detailed behavior description

The **`REPT DBCHG`** message reports any change in the system database. Changes can be caused by a TL1 user or by an internally-generated state change.

Changes in user provisionable attributes are reported as the equivalent TL1 command that would cause that change.

Changes in non-user provisionable attributes are reported using only the AID and the block parameters, i.e. leaving the *ccb* parameter empty.

System initialization

Any time a command is entered that causes a system reset, a **`REPT DBCHG`** message is transmitted before the system resets.

AID ranges

If a database change occurs as a result of a TL1 command in which an AID range is entered, the system responds separate **`REPT DBCHG`** messages for each affected AID.

Messages caused by own changes

If the database change occurs as a result of a TL1 command, a **REPT DBCHG** message is transmitted to users provisioned to receive **REPT DBCHG** messages, including the user which issued the command.

Example input/output

The following is an example of the **REPT DBCHG** command:

```
   UNITE-NE 00-11-08 08:00:00
A  001 REPT DBCHG
   ”TIME=07-59-59,DATE=00-11-08: ENT-EQPT:1-1-#-#-05::nvmname=OP2G5,nvmqual=13SR4”
;
```

References

GR-199-CORE

REPT EVT: Report event

REPT EVT messages are generated autonomously to report alarms of severity NA – Warning (ITU) / Not Alarmed (Telcordia). Also events like protection switches will be reported.

Output format

The general format of a **REPT EVT** autonomous message is as follows:

```
   sid date time
A  atag REPT EVT modifier

      ″aid:condtype,condeff,ocrdat,ocrtm,,,,[thlev][,tmper][:conddescr]″
   (1 or more lines of the above)
;
```

Note: The trailing comma has to be present if *thlev*, *tmper*, or both are used.

Since trailing commas to the left of a colon can be deleted, for non-TCA events, the output format is as follows:

```
   sid date time
A  atag REPT EVT modifier
   ″aid:condtype,condeff,ocrdat,ocrtm[:conddescr]″
   (1 or more lines of the above)
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **atag** – Automatic Tag
   This parameter is used for message sequencing. The number is incremented by one for each autonomous message sent by the NE.
   Type: Integer (3 digits).
   Range: 000 – 999 Leading zeros are not suppressed.
2. **modifier** – Message Modifier
   This parameter defines the AID type to which the alarm belongs.
   Type: Alphanumeric value set.
   Range:
   - COM: Common. An alarm/event that applies to the whole NE system; for example, a processing error.
   - EQPT: An equipment-related alarm/event.

- SYST: An alarm/event related to system timing.
- OUTT: An alarm/event related to output timing.
- STM1E: A facility-related alarm/event for synchronous transport mode 1 (electrical).
- STM1: A facility-related alarm/event for synchronous transport mode 1.
- STM4: A facility-related alarm/event for synchronous transport mode 4.
- STM16: A facility-related alarm/event for synchronous transport mode 16.
- STM16T: A facility-related alarm/event for fully transparent synchronous transport mode 16.
- STM64: A facility-related alarm/event for synchronous transport mode 64.
- STM256: A facility-related alarm/event for synchronous transport mode 256.
- ODU1: A facility-related alarm/event for optical data units of level 1
- OC3: A facility-related alarm/event at the optical carrier signal level 3.
- OC12: A facility-related alarm/event at the optical carrier signal level 12.
- OC48: A facility-related alarm/event at the optical carrier signal level 48.
- OC48T: A facility-related alarm/event at the fully transparent optical carrier signal level 48.
- OC192: A facility-related alarm/event at the optical carrier signal level 192
- OC768: A facility-related alarm/event at the optical carrier signal level 768
- T3: A facility-related alarm/event at the DS3 level
- TMAT3: A facility-related alarm/event at the DS3 level related to TransMUX function
- EC1: A facility-related alarm/event at the EC1 level
- 1GE: A facility-related alarm/event for 1 Gigabit Ethernet.
- 1GEPL: A facility-related alarm/event for 1 Gigabit Ethernet Private Line.
- 10GEPL: A facility-related alarm/event for 10 Gigabit Ethernet Private Line.
- VC3: A facility-related alarm/event for virtual container 3.
- VCG: A facility-related alarm/event for virtual concatenation group.
- VC4: A facility-related alarm/event for virtual container 4
- VC44C: A facility-related alarm/event for a signal concatenated at 4 times the VC4 rate.
- VC416C: A facility-related alarm/event for a signal concatenated at 16 times the VC4 rate.
- VC464C: A facility-related alarm/event for a signal concatenated at 64 times the VC4 rate.
- STS1: A facility-related alarm/event at synchronous transport signal level 1.

- TMASTS1: A facility-related alarm/event at synchronous transport signal level 1 related to TransMUX function.
- STS3C: A facility-related alarm/event at synchronous transport signal level 3.
- STS12C: A facility-related alarm/event at synchronous transport signal level 12.
- STS48C: A facility-related alarm/event at synchronous transport signal level 48.
- STS192C: A facility-related alarm/event at synchronous transport signal level 192.
- HOVC4 : A facility-related alarm/event for virtual container 4 higher-order path termination.
- HOSTS1: A facility-related alarm/event at synchronous transport signal level 1 higher-order path termination.
- LOVC3: A facility-related alarm/event for virtual container 3 (lower-order path).
- LOVC12: A facility-related alarm/event for virtual container 12
- LOVT1: A facility-related alarm/event at transport level VT1.5.
- TMSVT: A facility-related alarm/event at transport level VT1.5 related to TransMUX function.
- DS1: A facility-related alarm/event at transport level DS1.
- LINK: A signaling communication link related alarm/event for ONNS

3. **`aid`** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: All defined AIDs are allowed. It should be a valid AID for the specified *modifier*. Refer to the definition of AIDs.
4. **`condeff`** – Condition Effect
   This parameter indicates the effect of the reported event on the NE. It is used to distinguish between persistent and transient alarms/events.
   Two state transitions can be distinguished for a persistent alarm/event:
   - raise transition: state change from not present to present
   - clear transition: state change from present to not present.

   A transient alarm/event has no 'presence' state; i.e. just the occurrence of that event/alarm is reported. ABN condition are always persistent and not transient.
   Type: Alphanumeric value set.
   Range:
   - CL: Standing condition cleared for a persistent alarm/event.
   - SC: Standing condition raised for a persistent alarm/event
   - TC: Transient condition occurred for a transient alarms/events.

   Default: There is no default value.

5. **`conddescr`** – Description of the Condition
   ASAP related conditions:
   This is a textual Description of the Condition.
   Type: inner string of the format:
   \″<category>, <entity>, <description>, <additional text>\″ or
   \″<category>, <entity>, <description>, <additional text>-cleared\″ (this form is used in REPT ALM/EVT for cleared alarms)
   Range:
   - category: is the category
   - entity: is the entity for which the alarm is reported
   - description: is the Alarm description, the string is up to 32 characters
   - additional text: additional text.

   If there is no additional text, then the trailing comma may be omitted.
   TCA related conditions:
   The general format is: category, entity, description, additional text. If there is no additional text, then the trailing comma may be omitted.
   Type: inner string
   Range:
   - For STM<N>, STM<N>, OC<N>T, OC<N>: Quality of service, {rr} port, TCA where rr is equal to the port rate
   - For ODU<N>: Quality of service, {rr} path, TCA where rr is equal to the ODU rate (e.g., ODU1)
   - For VC3, VC4, VC44C, VC416C, VC464C, STS1, STS3C, STS12C, STS48C and STS192C: Quality of service, {rr} CS, TCA where rr is equal to the signal rate
   - For terminated VC4 and STS1signals: Quality of service, {rr} CS, TCA where rr is equal to the Constituent Signal (CS) rate
   - For GE port: Quality of service, {ge_rate}GE, TCA where ge_rate is the signal rate of the GE port
   - For VCG port: Quality of service, VCG, TCA
   - For DS3 port: Quality of service, T3 port, TCA.
   - For TMAT3 port: Quality of service, T3 port, TCA
   - For TMASTS1 path termination: Quality of service, {***rr***} path, TCA, where ***rr*** is equal to STS1 rate
   - For not terminated DS1 tributary: Quality of service, DS1 path, TCA

   Abnormal conditions:
   The general format is: category, description.
   Type: inner string.

Range:

- Description for a protection group AID: – Abnormal condition, forced switch active – Abnormal condition, lockout active
- Description for a timing reference AID: – Abnormal condition, operate in hold-over – Abnormal condition, operate in free run mode
- Description for a port AID: – Abnormal condition, facility loopback active
- Description for a logical tributary AID: – Abnormal condition, cross-connect loopback active – Abnormal condition, test access session active
- Description for a system AID: – Abnormal condition, System operates in maintenance mode – Abnormal condition, Autonomous event forwarding is inhibited.

Security related Conditions:
The parameter shall provide additional information w.r.t. the security event.
Type: Inner string
Range:
The following strings are applicable for the one remaining *condtype* SEC-USERLOGIN.
Also see the relation between *condeff* and *conddescr*:
condeff – conddescr
SC – \″login – <uid>\″
CL – \″logout – <uid>\″
CL – \″forced logout – <uid> \″
CL – \″inactivity timeout – \″
CL – \″logout, disconnected – <uid> \″
The security event notification is related to the user session which is indicated by the user ID (*uid*).

6. **`condtype`** – Condition Type
ASAP related conditions:
This parameter identifies the probable cause (type of alarm indication) being reported.
Type: Alphanumeric value set.
Range: All defined probable causes..
TCA related condition types:
This parameter identifies the type of event indication being reported.
Type: inner string with the Format: T-<montype>
Range:

- Range for SDH signals

  The following values are for all *STM<N> signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-MS-N-ES, T-MS-N-SES, T-MS-N-BBE, T-MS-N-UAS, T-MS-F-ES, T-MS-F-SES, T-MS-F-BBE, T-MS-F-UAS

  The following values are for all *STM<N>T signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-RS-N-ES-EGR, T-RS-N-SES-EGR, T-RS-N-BBE-EGR, T-RS-N-UAS-EGR

The following values also are for *STM64 and STM256 signals*: T-MS-N-FECC

The following values are for *VC3, VC4, VC44C, VC416C and VC464C tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, cVC-F-BBE, T-VC-F-UAS, T-PJCPOS, T-PJCNEG

The following values are for *VC12 tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, T-VC-F-BBE, T-VC-F-UAS

- Range for SONET signals

  The following values are for all *OC<N> and EC1 signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVL, T-ESL, T-SESL, T-UASL, T-CV-LFE, T-ES-LFE, T-SES-LFE, T-UAS-LFE

  The following values are for all *OC<N>T signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVS-EGR, T-ESS-EGR, T-SESS-EGR, T-SEFS-S-EGR

  The following values are also for *OC192 and OC768 signals*: T-FECC-L

  The following values are for STS1, STS3C, STS12C, STS48C and STS192C tributaries: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE, PPJC-PDet, NPJC-PDet, NPJC-PDet, NPJC-PGen

  The following values are for *TMASTS1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP, T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

  The following values are for *VT1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

- Range for DS3/DS1 signals

  The following values are for *DS3 signals with T3 modifier*:

  T-CVL, T-ESL, T-ESA-L, T-ESB-L, T-SESL, T-CVP, T-ESP, T-ESA-P, T-ESB-P, T-SESP, T-UASP, T-SAS-P, T-CV-PFE, T-ES-PFE, T-ESA-PFE, T-ESB-PFE, T-SES-PFE, T-UAS-PFE, T-SAS-PFE, T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS3 signals with TMAT3 modifier*:

  T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS1 signals*:

  T-CVP-EGR, T-ESP-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR

- Range for other signals

  The following values are for all *ODU<N> signals*:

  T-ODU-N-ES, T-ODU-N-SES, T-ODU-N-BBE, T-ODU-N-UAS, T-ODU-F-ES, T-ODU-F-SES, T-ODU-F-BBE, T-ODU-F-UAS

  For *ENET signals*:

  T-PDE, T-EDFE, T-EDFC

Abnormal condition types:
This parameter identifies the type of event indication being reported.
Type: inner string with the Format: ABN-<abntype>

Range:

- ABN-FORCESW: Forced switch active
- ABN-LOCKOUT: Lockout active
- ABN-HLDOVRSYNC: Holdover synchronization mode
- ABN-FREERUNSYNC: Free-run synchronization mode
- ABN-LPBKPORT : Facility Loopback
- ABN-LPBKXCON: Cross Connect Loopback
- ABN-MAINTENANCE: System operates in maintenance mode
- ABN-TESTACC: Test access session has been setup.

Security related Condition Types:
The parameter specifies/identify the type of security event being reported.
Type: Alphanumeric value set.
Range: SEC-USERLOGIN
Default: None.

7. **`ocrdat`** – Occurrence Date
This parameter indicates the raising / clearing date of the alarm.
Type: value set of the format MM-DD.
Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
8. **`ocrtm`** – Occurrence Time
This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
Type: value set of the format HH-MM-SS.
Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
9. **`thlev`** – Threshold Level
This parameter is only used for Threshold Crossing Alerts (TCAs). It reports the threshold level value which is set by ED-TCA-PROF.
The two applicable thresholding methods (tcamd) must be distinguished. For TR-only mode:
   - the raise TCA threshold value is reported.

   For TR-RTR mode:
   - the raise TCA threshold value is reported for the raise event
   - the reset TCA threshold value is reported for the clear event.

   Type: Integer.
Range: >1 (1 … 4294967295). Leading zeros will be suppressed.
10. **`tmper`** – Time Period
This parameter reports the monitored time interval. This parameter is only used for Threshold Crossing Alerts (TCAs).
Type: value set.
Range: 15-MIN, 1-DAY.

Detailed behavior description

If an alarm is provisioned as NA – Warning (ITU) / Not Alarmed (Telcordia), then the occurrence of that alarm is reported in a REPT EVT message. REPT EVT messages also occur when any such standing alarm clears and is used to report transient conditions.

A standing condition is an alarm event that has a beginning and an end. When the event begins, the standing condition is declared via an autonomous message. When the event ends, an autonomous message is sent to indicate that the event cleared.

A transient condition only results in one autonomous message – a declaration that a specific transient condition occurred.

Autonomous performance monitoring (PM) threshold crossing alert (TCA) messages are provided to the OS using the REPT EVT TL1 message.

1675 LambdaUnite MSS supports two thresholding methods:

1. TR-only (also known as the implicit reset method). Then the TCA is a transient one – it is reported when the threshold is crossed. Because it is transient, there is no report when it clears (TC only).
2. TR-RTR (also known as the explicit reset method). Then the TCA has a standing condition, this means it has a beginning and an end (SC and CL).

The severity of a TCA is not provisionable. It is always NA – warning (ITU) / not alarmed (Telcordia) which implies TCA messages are always reported via REPT EVT.

Protection switching messages are provided to the OS using the REPT EVT TL1 message.

The REPT EVT message reports updates of the protection states.

Abnormal Condition messages are provided to the OS using the REPT EVT TL1 message.

The REPT EVT message reports updates of the abnormal condition states. The abnormal condition is always a standing condition, this means it has a beginning and an end (SC and CL).

The severity of a ABN condition is not provisionable. It is always NA – warning (ITU) / not alarmed (Telcordia) which implies ABN messages are always reported via REPT EVT.

Notification of Abnormal Condition Test Access session active

The system shall notify the Abnormal Condition “Test Access session active” for the tributaries under test, i.e. the E and F tributaries. For each tributary under test, one notification is sent.

The tributary AID and the number of notifications depends on the Test Access mode.

For single test tributary modes one notification is sent:

- MONE/SPLTE – one notification for the E tributary
- MONF/SPLTF – one notification for the F tributary

For dual test tributary modes two notifications are sent:

- MONEF/SPLTEF – one notification for the E tributary and one for the F tributary

Example input/output

The following example shows an autonomous message REPT EVT for an LOS Alarm which has a severity NA assigned:

```
   LT-UNITE-123 00-01-01 08:00:00
A  003 REPT EVT OC48
   ″1-1-#-#-3-1:AIS-L,SC,01-01,07-59-59:\″Communication-Transport, OC48 port, Alarm
      Indication Signal\″″
;
```

The following example shows an autonomous message REPT ALM for an LOS Alarm clear which has a severity NA assigned:

```
   LT-UNITE-123 00-01-01 08:00:00
A  003 REPT EVT OC48
   ″1-1-#-#-3-1:AIS-L,CL,01-01,07-59-59:\″Communication-Transport, OC48 port, Alarm
      Indication Signal-cleared\″″
;
```

Note: Examples for the other types of REPT EVT e.g. TCAs and protection switches are described in other sections which describes the specifics of the area to which the events belongs to (e.g. performance monitoring, protection switching).

Related TL1 messages

- "ALW-MSG: Allow message" (p. 3-33)
- "ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
- "INH-MSG: Inhibit message" (p. 3-2021)
- "REPT ALM: Report alarm" (p. 3-2096)
- "REPT-ALM-ENV: Report alarm environment" (p. 3-2101)
- "RTRV-ALM-ENV: Retrieve alarm environment" (p. 3-2238)
- "RTRV-COND: Retrieve condition" (p. 3-2285)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: NE and Transport Surveillance Messages.

REPT EVT FXFR: Report event FXFR

REPT EVT FXFR is an autonomous message that is used to report remote file transfer activities.

Output format

The general format of a **REPT EVT FXFR** autonomous message is as follows:

```
   sid date time
A  atag REPT EVT FXFR
                              ″filename,fxfr_status[,fxfr_rslt][,bytes_xfrd]″;
```

Output parameters

Command parameters

1. **atag** – Automatic Tag
   This parameter is used for message sequencing. The number is incremented by one for each autonomous message sent by the NE.
   Type: Integer (3 digits).
   Range: 000 – 999 Leading zeros are not suppressed.
2. **filename**
   This parameter identifies the name of the file transferred.
   Type: text string.
   Range: 1 to 128 characters.
3. **fxfr_status**
   This parameter indicates the file transfer status.
   Type: Alphanumeric value set.
   Range:
   - START, indicates file transfer has started
   - IP, indicates file transfer is In-Progress; optional
   - COMPLD, indicates file transfer has completed.
4. **fxfr_rslt**
   This parameter indicates success or failure of the file transfer. This parameter is only displayed when the file transfer has completed (that is., *fxfr_status* is COMPLD).
   Type: Alphanumeric value set.
   Range:
   - SUCCESS, indicates successful file transfer
   - FAILURE, indicates failed file transfer
5. **bytes_xfrd**
   Number of bytes transferred for IP and total number of bytes for COMPLD.
   Type: Alphanumeric value set.

Detailed behavior description

After the initial response to the **COPY-RFILE** command by the NE, the autonomous message must be used to communicate any kind of event. So, if the connection or file transfer fails, the autonomous message must be sent with a failure indication. Once the file has been completely downloaded, the autonomous message must then be sent with a successful indication.

Reasons of transmit failure - external communication failure

If the command failed due to an external communications failure.

The following list identifies conditions that will cause an SROF error response as given above:

- External communication failure
- Unsuccessful file transfer.

Reasons of transmit failure - transferred file is invalid

If the transferred file is deemed invalid for miscellaneous reasons such as incorrect checksum, invalid control information, etc., the system responds with *fxfr_rslt* = FAILURE.

Backup failed after unsuccessful retransmission of files

If during a database backup a transaction log overflow occurs, and the retransmission of database files fails, then the system responds with *fxfr_rslt* = FAILURE.

During database backup, all database changes caused by provisioning activities are stored temporarily in a transaction log. This prevents changes of the database while the backup is running. Excessive provisioning during database backup may cause an overflow of the transaction log. In this case, the changes are written to the database and the affected database files are retransmitted. In the case that the retransmission fails, the backup is terminated with error.

Reasons of transmit failure - Generic not available

If the command failed because the generic is not available, the system shall respond with fxfr_rslt=FAILURE. The following condition will cause this ENSG error response: Not a software generic file.

Reasons of transmit failure - Directory not accessible

If a directory is specified that cannot be found or cannot be accessed due to improper privileges, the system responds with *fxfr_rslt* = FAILURE.

Reasons of transmit failure - Product mismatch

If the command failed due to a product mismatch in a retrieved file (file does not belong to NE type), the system responds with *fxfr_rslt* = FAILURE.

ES64U pack mismatch

If the command failed due to one of the following ES64U pack mismatches:

- A database of a available ES64U cannot retrieved and is therefore not contained in the uploaded database generic (Backup).
- A downloaded database generic contains ES64U databases which have a mismatch with the expected databases.

In all cases, the system responds with *fxfr_rslt* = FAILURE.

Example input/output

The following example shows an autonomous message **REPT EVT FXFR** for starting a file transfer:

```
   LT-UNITE-123 00-01-01 08:00:00
A  003 REPT EVT FXFR
   ″/UNITE/release4,START,8051″
;
```

The following example shows an autonomous message **REPT EVT FXFR** for an inprogress statement:

```
   LT-UNITE-123 00-01-01 08:00:00
A  003 REPT EVT FXFR
   ″/UNITE/release4,IP,2000″
;
```

The following example shows an autonomous message **REPT EVT FXFR** for a successful file transfer:

```
   LT-UNITE-123 00-01-01 08:00:00
A  003 REPT EVT FXFR
   ″/UNITE/release4,COMPLD,SUCCESS,8051″
;
```

Related TL1 messages

- "APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
- "COPY-RFILE: Copy remote file" (p. 3-106)
- "RTRV-PRMTR-DATA: Retrieve parameter data" (p. 3-2744)
- "RTRV-PRMTR-SFTWR: Retrieve parameter software" (p. 3-2753)
- "RTRV-NE: Retrieve NE" (p. 3-2549)
- "ED-NE: Edit NE" (p. 3-442)

REPT L1ROUTING notification

Overview

The **REPT L1ROUTING** notification is used to notify about TL1 triggered (ACT/CANC-L1ROUTING) enabling or disabling of a level 1 Routing Controller.

The **REPT L1ROUTING** notification message reports if an ONNS node is enabled or disabled as level 1 Routing Controller. Enabling or disabling are caused by a user's TL1 command.

Notification messages resulting from a TL1 command that were caused by the user's own requested changes

If an ONNS node is enabled or disabled as level 1 Routing Controller as a result of a TL1 ACT-L1ROUTING or CANC-L1ROUTING command, then a REPT L1ROUTING message is transmitted to all users provisioned to receive REPT L1ROUTING messages, including the user that issued the command.

Therefore, a management OS must expect to receive messages for changes it caused as well as those made by a CIT user.

Output format

Output format for a REPT L1ROUTING notification message after ACT-L1ROUTING

The general format of a **REPT L1ROUTING** notification is as follows:

```
   srcTid date time
A  atag REPT L1ROUTING
   ″TIME=ocrTime,DATE=ocrDate:RCStatus=rcStatus:srcASTNNodeID″
   ″L1Routing=routeFunc,L1RoutingAreaID=routeAreaID,L1ParentRoutingAreaID=
prouteAreaID,L1RoutingControllerID=routeControlID,L1RoutingControllerIPAddr=
routeControlIPAddr,AbstractModel=abstractModel,AbstractNodeID=abstractNodeID″;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date
3. **`time`**
   Time. The time parameter is set to the time when the output message is generated.
   See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time
4. **`atag`** – Automatic Tag
   The *atag* is used for message sequencing. The *atag* is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Type: decimal number
   Range: 000-999
   Default: 0
5. **`ocrDate`** – Occurence Date
   Date. The date parameter is set to the date when the event occurred in the system.
   Type: Value set.
   Range:

YY-MM-DD, where

- YY=00-99,
- MM=01-12,
- DD=01-31

Default: occurence date

6. **`ocrTime`** – Occurence Time
   Time. The time parameter is set to the time when the event occurred in the system.
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: occurrence time
7. **`rcStatus`** – Routing Controller Status
   Represents the change of The ONNS node as level 1 Routing Controller caused by a TL1 command.
   Type: Value set.
   Range:
   - *enabled*: ONNS node is enabled as level 1 Routing Controller
   - *disabled*: ONNS node is disabled as level 1 Routing Controller
8. **`ASTNNodeID`**
   ONNS source nodeID. Together with the following *srcPortID* field defines the associated Ingress *connPoint* that connects the source node to the edge/gateway switch or client router.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0..255
9. **`routeFunc`**
   routeFunc defines if the *LSAAdvertisedRouter* has level 1 routing controller functionality.
   Type: Alphanumeric value set.
   Range: PRIMARY (ONNS node is the primary level 1 routing controller)
10. **`routeAreaID`**
    routeAreaID defines the identity of the routing area.
    Type: four dot separated numeric string
    Range: 0 – 255 for each numeric string.
11. **`prouteAreaID`**
    prouteAreaID defines the identity of the parent routing area.
    Type: four dot separated numeric string
    Range: 0 – 255 for each numeric string.

12. **`routeControlID`**
    routeControlID defines the identity of the level 1 routing controller.
    Type: four dot separated numeric string
    Range: 0 – 255 for each numeric string.
13. **`routeControlIPAddr`**
    routeControlIPAddr defines the IP address for this routing controller node as level 1 routing controller address.
    Type: four dot separated numeric string
    Range:
    For first numeric string: 0 … 223
    For second to fourth numeric string: 0 … 255
    The values 0.0.0.0 and 1.1.1.1 are excluded.
14. **`abstractModel`**
    abstractModel defines the abstraction model used by the level 1 routing controller.
    Type: Alphanumeric value set.
    Range: ABSTRACTNODE
15. **`abstractNodeID`**
    abstractNodeID is defined in case the *abstractModel* is ABSTRACTNODE. This parameter specifies the abstract node ID, which is advertised in the level 1 area.
    Type: four dot separated numeric string
    Range: 0 – 255 for each numeric string
    Parameter *abstractNodeID* is optional for *abstractModel* values other then ABSTRACTNODE.

Example output

Output example for an enabled level 1 routing controller

The following example shows the output of a **`REPT L1ROUTING`** notification caused by an **`ACT-L1ROUTING TL1`** command:

```
   LT-ONNS9 2002-07-18 08:00:00
A  113456 REPT DBCHG
   ”TIME=07-59-59,DATE=2002-08-01:ACT-L1ROUTING:::RoutingFunc=PRIMARY,
RoutingAreaID=111.234.215.6,ParentRoutingAreaID=111.214.115.7,
L1RoutingControllerID=218.3.45.56,L1RoutingControllerIPAddr=129.32.19.2,
AbstractModel=ABSTRACTNODE,AbstractNodeID=129.239.4.23”;
```

Related TL1 messages

- “ACT-L1ROUTING” (p. 3-15)
- “CANC-L1ROUTING” (p. 3-54)

REPT-NNCHG notification

Overview

The **REPT NNCHG** notification is used to notify about NNDR bandwidth availability changes. In addition, the **REPT NNCHG** notification is used to notify about ONNS restarts. In addition the **REPT NNCHG** notification is used to notify about ONNS System Parameter changes by **ED-NNCNFG** command. In addition, the **REPT NNCHG** notification is used to notify about ONNS Defragmentation Operation Completion by **OPR-DEFRAG** command.

The **REPT NNCHG** notification message reports NNDR bandwidth availability changes which trigger distribution of a new Link State Advertisement (LSA). In addition, the **REPT NNCHG** notification message reports restarts of the ONNS which in case of the XC would not become transparent for the connected management system, otherwise. This notification then would allow a management system to execute **RTRV-PATH** commands to the affected node to align on connection data and states. This is necessary, as a management system is neither notified about connection data and state changes during downtime of the XC nor about actual connection data and states after recovery of the XC. In addition the **REPT NNCHG** notification message reports the change of the ONNS System Parameters like “wait to restore timer”, “default restoration priority”, or “administrative cost escalation”.

Output format

Output format for REPT NNCHGfor ONNS data repositories

The generic format of a **REPT NNCHG** message is as follows with the update management block (*umb*) and the common data block (*comBlock*).

```
   srcTid rpt-date rpt-time
A  atag REPT NNCHG
   ″umb:comBlock″
;
```

Expanded output format for REPT NNCHG for ONNS data repositories

The general format of a **REPT NNCHG** autonomous notification message with expanded *umb* and *comBlock* is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT NNCHG
   ″TIME=ocrTime,DATE=ocrDate:srcNodeID:srcAid:dataSrcID:msgCode,,
\″ eventText \″,[eventData]″
;
```

Output parameters

Command parameters

1. **`srcTid`** – Source Node Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`rpt-date`** – Report date
   This parameter is set to the date when the output message is generated.
   Note: Format uses a 4 digit year instead of a 2 digit year as for host TL1 commands. Compare to RTRV-HDR.
   Range: YY-MM-DD; where YY=2000-2099, MM=01-12, DD=01-31.
3. **`rpt-time`** – Report time
   This parameter is set to the time when the output message is generated.
   Range: HH:MM:SS; where HH=00-23, MM=00-59, SS=00-59.
4. **`atag`** – Automatic Tag
   This parameter is used for message sequencing. The atag is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Values: 000 (default) – 999
5. Update Management Block (umb): **`ocrDate`** – Occurrence Date
   The date parameter is set to the date when the event occurred in the system.
   Values: .YY-MM-DD where YY=2000-2099, MM=01-12, DD=01-31
6. umb: **`ocrTime`** – Occurrence Time
   Time when the change occurred.
   The time parameter is set to the time when the event occurred in the system.
   Values: HH-MM-SS where HH=0-23, MM=00-59, SS=00-59.
7. Common Block (cb): **`srcNodeID`**
   ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
   Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
8. cb: **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
9. cb: **`msgCode`**
   This parameter identifies the reported change.
   Type: Alphanumeric string of length.

10. cb: **`dataSrcID`**
    The data repository or software package reporting the change: NNDR for the Network Neighbor Data Repository.
    Range:
    - NNDR: for the Network Neighbor Data Repository
    - NNSYS: for ONNS System
    - NNDEFRAG: for the Defragmentation Operation
11. cb: **`eventText`**
    Free format text, up to 80 characters.
12. cb: **`eventData`**
    Further event data as named-parameter=value pair(s). Comma separated when multiple pairs.

Example output

`REPT NNCHG` output example for threshold-triggered LSA notification

The following is an example of the **`REPT NNCHG`** message for an ONNS Neighbor Nodal Data Repository LBA update due to a capacity threshold crossing, which causes an LSA distribution:

```
   LT-ONNS-101 2002-08-01 08:02:37
A  593 REPT NNCHG
   ″TIME=08-02-00,DATE=02-08-01:178.223.45.172:SNN:NNDR:C050,,\″ LSA update sent due
       to significant bandwidth change
\″,TC=50,,NeighborNodeID=178.23.145.12″
;
```

`REPT NNCHG` output example for timer-triggered LSA notification

The following is an example of the **`REPT NNCHG`** message for an Neighbor Nodal Data Repository LBA update due to timer expiration, which causes an LSA distribution:

```
   LT-ONNS-007 2002-08-01 08:02:05
A  152 REPT NNCHG
   ″TIME=08-02-00,DATE=02-08-01:178.223.45.172:SNN:NNDR:C060,,\″ LSA update sent due
       to periodic timer expiry \″,TMR=15″
;
```

Note: This message only occurs when the capacity changes have not been reported by a capacity threshold-triggered LSA notification.

`REPT NNCHG` output example for ONNS restart

The following is an example of the **`REPT NNCHG`** message sent after a restart of the ONNS:

```
   LT-ONNS-007 2003-07-01 08:02:05
```

```
A  152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNSYS:C085,,\″ NN Ready  \″″
;
```

The management system needs this information to perform a resynchronization for example in case of an XC reboot with link bundle IDs being new assigned.

REPT NNCHG output example for ONNS restart initiation

The following is an example of the **REPT NNCHG** message sent when a restart of the ONNS applications has been initiated manually:

```
   LT-ONNS-007 2003-07-01 08:02:05
A  152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNSYS:C080,,\″ NN Restart  \″″
;
```

An ONNS restart can be initiated manually by executing a specific **TST-PKG** TL1 command. This TL1 command is not applicable in general, but its use is restricted to authorized persons only. This notification is especially important to document manually initiated ONNS restarts within notification logs and to distinguish such restarts from autonomous ones.

REPT NNCHG output example for ONNS System Parameter changes

The following is an example of the **REPT NNCHG** message sent after a change of ONNS System Parameters :

```
ED-NNCNFG:LT-ONNS-A::123456::waitToRevertTimer=120,defaultRstrPriority=4;

   LT-ONNS-A 2003-07-01 08:02:05
A  152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNSYS:C090,,\″ NN System
      Parameter Changed and Stored in NVM
      \″,WaitToRevertTimer=120,DefaultRstrPriority=4″
;
```

The event data shall contain all changed ONNS system parameters (see output specblock parameters of ED-NNCNFG)

REPT NNCHG output example for a successful defragmentation operation

The following is an example of the **REPT NNCHG** message sent after successful defragmentation operation:

```
OPR-DEFRAG:LT-ONNS-A::123456::LINK,29.13.80.192;

   LT-ONNS-A 2003-07-01 08:02:05
A  152 REPT NNCHG
```

```
   "TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C150,,\" Defragmentation
       successfully completed \",LINK,TargetID=29.13.80.192"
   "MovedPaths:135.13.19.203-43216,135.13.198.103-41216,135.13.194.233-43226,
135.13.106.212-42916,135.13.139.203-43216,135.13.129.203-43216"
   "MovedPaths:135.13.196.217-43216,135.13.196.203-43319,135.13.139.198-43216,
135.13.146.203-43216,135.13.199.203-43216,135.13.196.213-43216"
...
  "MovedPaths:135.13.196.203-40216,135.13.196.203-43916,135.13.196.263-43216,
135.13.135.203-43216,135.13.196.218-43216,135.13.196.203-44156"
;
```

This event is sent out after successfully intra port, inter port or empty port operation

REPT NNCHG output example for an interruption of the defragmentation operation

The following is an example of the **REPT NNCHG** message sent after interruption of the defragmentation operation:

```
OPR-DEFRAG:LT-ONNS-A::123456::LINK,29.13.80.192;

   LT-ONNS-A 2003-07-01 08:02:05
A  010152 REPT NNCHG
   "TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C151,,\" Defragmentation
       partially completed: Local Bridge and Merge Failure
\",LINK,TargetID=29.13.80.192,PathID=135.13.196.203-432165"
    "MovedPaths:135.13.196.203-40216,135.13.196.203-43916,135.13.196.263-43216,
135.13.135.203-43216,135.13.196.218-43216,135.13.196.203-44156"

;
```

There could be situations in which the defrag operation gets interrupted or terminated such that it was not able to cleanly complete the particular connection move (bridge and merge on the local or remote end of the connection) that it was involved in.

In these cases it is possible that a dangling bridge and merge cross-connection is left on either the local or the remote end. When this happens, an additional value is returned as part of the qualifier, that is the *pathID* of the connection move that was left incomplete.

The following list identifies conditions that will cause a similar notification:

- C151 (LBMF) Local Bridge and Merge Failure, [Scope,TargetID,PathID]
- C152 (LXCF) Local Cross Connect Release Failure, [Scope,TargetID,PathID]
- C153 (RBMF) Remote Bridge and Merge Failure, [Scope,TargetID,PathID]
- C154 (RXCF) Remote Cross Connect Release Failure, [Scope,TargetID,PathID]
- C155 (LDPD) Path Being Deleted, [Scope,TargetID]
- C156 (NNRP) No Neighbor Response, [Scope,TargetID,PathID]

Notification C153 is also send if a port of the remote side which should be defragmented is in LOCKOUT state.

REPT NNCHG output example for an incomplete empty port operation

The following is an example of the **REPT NNCHG** message sent after incomplete empty port operation

```
OPR-DEFRAG:LT-ONNS-A::123456::EMPTYPORT,1-1-1-2;

   LT-ONNS-A 2003-07-01 08:02:05
A  010152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C157,,\″ Empty Port
      partially completed \″,EMPTYPORT,TargetID=1-1-1-2,RemainPaths=2″
   ″MovedPaths:135.13.196.203-40216,135.13.196.203-43916,135.13.196.263-43216,
135.13.135.203-43216,135.13.196.218-43216,135.13.196.203-44156″

;
```

The port has inmovable paths or there is not enough space in other ports of the link bundle to completely empty the port.

REPT NNCHG output example for a collision failure of the defragmentation operation

The following is an example of the **REPT NNCHG** message sent after collision or communication failure of the defragmentation operation: The node on the far end of the fiber is also performing a defrag action on the same fiber. Or, the local node is performing a defrag action on a separate fiber with both actions affecting the same path. Or the node on the far end is performing a defrag action on a separate fiber with both actions affecting the same path. In this case the operate defrag returns without any action performed.

```
OPR-DEFRAG:LT-ONNS-A::123456::LINK,29.13.80.192;

   LT-ONNS-A 2003-07-01 08:02:05
A  010152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C158,,\″ Defragmentation
       unsuccessful: Conflicting Action at Local or Neighbor Node
\″,LINK,TargetID=29.13.80.192″
;
```

REPT NNCHG output example if a HW failure occurs during defragmentation operation

The following is an example of the **REPT NNCHG** message sent after a HW failure occurs during the defragmentation operation:

```
OPR-DEFRAG:LT-ONNS-A::123456::LINK,29.13.80.192;

   LT-ONNS-A 2003-07-01 08:02:05
A  010152 REPT NNCHG
```

```
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C159,,\″ Defragmentation
       unsuccessful: Port or Line in Failed State
\″,LINK,TargetID=29.13.80.192″
;
```

In the rare case that a catastrophic failure occurs (local node crashes, remote node crashes, the fiber gets cut) while a defrag operation is in progress, the operation will return with a defragmentation unsuccessful notification.

REPT NNCHG output example if a SW failure occurs during defragmentation operation

The following is an example of the **REPT NNCHG** message sent after a SW failure occurs during the defragmentation operation:

```
OPR-DEFRAG:LT-ONNS-A::123456::LINK,29.13.80.192;

   LT-ONNS-A 2003-07-01 08:02:05
A  010152 REPT NNCHG
   ″TIME=08-02-00,DATE=03-07-01:178.223.45.172:SNN:NNDEFRAG:C160,,\″ Defragmentation
       denied: SW failure \″,LINK,TargetID=29.13.80.192″
;
```

In the rare case that a software failure occurs while a defrag operation is in progress, the operation will return with a DENY notification.

Related TL1 messages

- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “REPT-NNEVT notification” (p. 3-2144)
- “RTRV-NNDR: Retrieve ONNS neighbor nodal data repository” (p. 3-2561)
- “RTRV-NTDR: Retrieve ONNS network topology data repository” (p. 3-2588)

REPT-NNEVT notification

Overview

REPT NNEVT notification messages are generated autonomously to report internal ONNS events and status changes such as signaling session failures, internal communication problems, and NVM errors, and Threshold Control Alert status changes..

Output format

The generic format of a **REPT NNEVT** autonomous notification message is as follows with the update management block (*umb*) and the common data block (*comBlock*):

```
   srcTid rpt-date rpt-time
A  atag REPT NNEVT
   ″umb:comBlock″
;
```

Expanded output format

The general format of a REPT NNEVT autonomous notification message with expanded *umb* and *comBlock* is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT NNEVT

      ″TIME=ocrTime,DATE=ocrDate:srcNodeID:srcAid:msgCode:dataSrcID:condType:condDescr,\″ event
;
```

Output parameters

Command parameters

1. **srcTid** – Source Node Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **rpt-date** – Report date
   This parameter is set to the date when the output message is generated.
   Note: Format uses a 4 digit year instead of a 2 digit year as for host TL1 commands. Compare to RTRV-HDR.
   Range: YY-MM-DD; where YY=2000-2099, MM=01-12, DD=01-31.

3. **`rpt-time`** – Report time
   This parameter is set to the time when the output message is generated.
   Range: HH:MM:SS; where HH=00-23, MM=00-59, SS=00-59.
4. **`atag`** – Automatic Tag
   This parameter is used for message sequencing. The atag is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Values: 000 (default) – 999
5. umb – Update Management Block
   - **`ocrDate`** – Occurrence DATE
     The date parameter is set to the date when the event occurred in the system.
     Values: .YY-MM-DD where YY=2000-2099, MM=01-12, DD=01-31
   - **`ocrTime`** – Occurrence TIME
     The time parameter is set to the time when the event occurred in the system.
     Values: HH-MM-SS where HH=0-23, MM=00-59, SS=00-59.
6. comBlock – Common Block
   This parameter contains the following fields:
   - **`srcNodeID`**
     ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
     Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
   - **`srcAid`**
     Source node ONNS application access identifier of the ONNS controller OS command interface port.
     Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
   - **`msgCode`**
     This code identifies the reason for the event:
     Type: Alphanumeric string of length 4.
   - **`dataSrcID`**
     The software package originating the event:
     CONN (for Connection Data Mgmt), H-INTF (for Host Interface Pkg), NNDM (Network Neighbor Data Mgmt), NN-SIG (for Signaling Pkg), NNSYS (for ONNS System Pkg).
   - **`condType`** – Condition Type
     Values: GEV (General Event), HBO (Host Buffer overflow), NVF (NVM Failure), SCE (Signaling Comm Established), SCF (Sig Comm Failure), TMO (TimeOut expired), TCA (Threshold Control Alert status change).
   - **`condDescr`** – Description of the condition
     This parameter consists of the following comma separated fields: category, entity, description code.

Category values: SIGSES = SIG session; COMM = Communications; STATUS = Event Status

Entity values: NVM, SYSTEM, DCNPORT, LINKBUNDLE

Description code: FAILED = status failure; OK = status okay; ERROR = error condition, INACTIVE = status value, LOW = status value, HIGH = status value.

- **`eventText`**

  Free format text, up to 80 characters.

- **`eventData`**

  Event data, as name-value pairs.

  For *condType* = TCA, the following name-value pairs are used: *LinkId* = aSTNNodeId&neighborASTNNodeID, *LinkBundleID* = linkBundleID.

Example output

The following **`REPT NNEVT`** autonomous message illustrates how an ONNS failure event is reported for signaling communication loss.

```
   LT-WS-ONNS-007 2002-08-01 08:00:00
A  00359 REPT NNEVT
   ″DATE=2002-08-01,TIME=07-59-59:135.13.196.45:SNN:F401:NN-
      SIG:SCF:SIGSES,DCNPORT,ERROR,\″ SIG Communications Loss
      \″,NBRID=neighborNodeID″
;
```

Other failure notification messages are listed below:

- F403 – CONN:NVF:COMM, NVM, ERROR – CONN NVM write error
- F404 – H-INTF:HBO:COMM, HostPort, ERROR – Host comm buffer overflow error
- F405 – NNSYS:GEV:STATUS,SYSTEM, FAILED – NN Failed
- F406 – NNDM:NVF:COMM, NVM, ERROR – NNDM NVM write error
- F407:NNDM:XCF:XCON,NNIPORT,FAILED,An XConn Failure occurred during Bridge to original Revertive path connection
- F408:NNDM:XCF:XCON,NNIPORT,FAILED, An XConn Failure occurred on Merge to Revertive path connection
- F409:NNDM:XCF:XCON,TSTPORT,FAILED, Test port bridge XCON failed
- F411:NNDM:EQF:PRTST,TSTPORT,FAILED,Test port equipment failed.

`REPT NNEVT` for clearing of a failure

An ONNS clearing event notification is sent when a failure condition has been repaired or cleared. The following event indicates successful establishment of a signaling communication session. Some ONNS failures do not have clearing notifications:

```
   LT-WS-ONNS-007 2002-08-01 08:00:00
```

```
A   00359 REPT NNEVT
   "DATE=2002-08-01,TIME=07-59-59:135.13.196.45:SNN:S451:NN-SIG:SCE:SIGSES,
      DCNPORT,OK,\" SIG Communication Session Established \",NBRID=198.168.224.37"
;
```

ONNS TCA events shall have a status message code that begins with S (i.e. Sxxx).

REPT NNEVT for raising TCA

The following **REPT NNEVT** autonomous message illustrates how an ONNS TCA event is reported for the TCA status becoming LOW:

```
   LT-WS-ONNS-007 2002-08-01 08:00:00
A   00359 REPT NNEVT
   "DATE=2002-08-01,TIME=07-59-59:135.13.196.45:SNN:S502:NNDM:TCA:
      STATUS,LINKBUNDLE,LOW,\" LOW status entered
      \",LinkID=135.13.196.45&135.13.25.68,LinkBundleID=5"
;
```

Other TCA related notification messages (including S502) are listed below:

- S501 – NNDM:TCA:STATUS,LINKBUNDLE,INACTIVE, INACTIVE status entered
- S502 – NNDM:TCA:STATUS,LINKBUNDLE,LOW, LOW status entered
- S503 – NNDM:TCA:STATUS,LINKBUNDLE,HIGH, HIGH status entered

REPT NNEVT for clearing TCA

The following **REPT NNEVT** autonomous message illustrates how an ONNS TCA event is reported for the clearing of a TCA:

```
   LT-WS-ONNS-007 2002-08-01 08:00:00
A   00359 REPT NNEVT
   "DATE=2002-08-01,TIME=07-59-59:135.13.196.45:SNN:S501:NNDM:TCA:
      STATUS,LINKBUNDLE,INACTIVE,\" INACTIVE status entered
      \",LinkID=135.13.196.45&135.13.25.68,LinkBundleID=5"
;
```

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)

- “REPT-NNCHG notification” (p. 3-2137)
- “RTRV-NNEVT-TCA” (p. 3-2582)

REPT-PTHCHG notification

Overview

The **REPT PTHCHG** notification is used to notify about TL1 triggered changes (**ENT/ED/DLT-PATH**) or autonomously triggered changes, such as restoration, of ONNS connection parameters.

The **REPT PTHCHG** notification is used to notify about UNI triggered changes (create/delete connections) of ONNS connection parameters.

The **REPT PTHCHG** notification is used to notify about Network Connection TL1 triggered changes (**ENT/ED/DLT-CONN**) of ONNS connection parameters.

The **REPT PTHCHG** notification message reports specific changes in the ONNS system data repositories. Changes can be caused by a user's TL1 path command, by UNI connection requests or by autonomous internally-generated connection changes such as path restoration.

Notification messages resulting from a TL1 command that were caused by the user's own requested changes

If the data repository change occurs as a result of a TL1 ent-path or dlt-path or ed-path-prot/roll command, then a REPT PTHCHG message is transmitted to all users provisioned to receive REPT PTHCHG messages, including the user that issued the command. Therefore, a management OS must expect to receive data repository change messages for changes it caused as well as those made by a CIT user. The REPT PTHCHG CCB field will indicate the TL1 command that caused the connection change.

Notification messages resulting from UNI connection requests

If the data repository change occurs as a result of a UNI connection setup or delete request, then a REPT PTHCHG message is transmitted to all users provisioned to receive REPT PTHCHG message. There will be separate PTHCHG messages from every domain with ONNS functionality affected by the connection request. Therefore, a management OS must expect to receive data repository change messages for changes caused by UNI connection requests. The REPT PTHCHG CCB field will indicate the UNI connection requests by the corresponding TL1 commands ENT-PATH and DLT-PATH.

Notification messages resulting from a TL1 network connection request

If the data repository change occurs as a result of a TL1 ENT-CONN or DLT-CONN, then a REPT PTHCHG message is transmitted to all users provisioned to receive REPT PTHCHG messages. There will be separate PTHCHG messages from every domain with ONNS functionality affected by the connection request. Therefore, a management OS must expect to receive data repository change messages for changes caused by TL1 network connection requests. The REPT PTHCHG CCB field will indicate the TL1 network connection requests by the corresponding TL1 commands ENT-PATH and DLT-PATH.

REPT PTHCHG for path/connection object creation

Only the connection objects (identified by *connID* and *pathID's*) generated by the TL1 ENT-PATH setup commands or a UNI connection request are returned in a REPT PTHCHG message. All of the supporting objects (such as cross-connections that are reported by the host NE) which are autonomously created due to the original connection object's creation are not reported in a ONNS REPT PTHCHG notification message. The Host NEs are responsible for sending REPT-DBCHG's for each cross-connection along the path of the connection.

REPT PTHCHG for path/connection object creation

Only the connection objects (identified by *connID* and *pathID's*) indirectly generated by the TL1 ENT-CONN setup commands or a eNNI connection request are returned in a REPT PTHCHG message. All of the supporting objects (such as cross-connections that are reported by the host NE) which are autonomously created due to the original connection object's creation are not reported in a ONNS REPT PTHCHG notification message. The Host NEs are responsible for sending REPT-DBCHG's for each cross-connection along the path of the connection.

TL1 commands associated with multiple *connID*'s

If a data repository change occurs as a result of a TL1 command in which multiple connections are involved, then the ONNS system will output separate REPT PTHCHG notification messages for each affected *connID*. The REPT PTHCHG CCB field will indicate the TL1 command that caused the connection change.

UNI connection requests associated with multiple *connID*'s

If a data repository change occurs as a result of a UNI connection request in which multiple connections are involved (e.g. Ethernet UNI connection), then the ONNS system will output separate REPT PTHCHG notification messages for each affected *connID*. The REPT PTHCHG CCB field will indicate the corresponding TL1 command of the connection request that caused the connection change.

The REPT PTHCHG message contains *pst*, *sst* and *tst* state information

The primary pathState (*pst*), secondary state (*sst*) and optional tertiary state (*tst*) parameters of a connection and path appear in the REPT PTHCHG message just as reported in the connection record that is returned in a TL1 command response. If a path or connection's state is changed by a TL1 command, the *pst*, *sst* and *tst* parameters that are returned in the REPT PTHCHG message are the actual *pst*, *sst* and *tst* parameters of the connection object after the state change has occurred. When reporting a path restoration change, the state included in the REPT PTHCHG message is the *ConnStatus* at the time that the message is sent.

The REPT PTHCHG message associated with a connection always contains the entire connection record and includes path Direction (*pdir*). If a connection object's configuration changes autonomously, the *pst*, *sst* and *tst* parameters of the REPT PTHCHG are the actual *pst*, *sst*, and *tst* parameters of the connection object after the configuration change has occurred.

If a connection status or path state changes autonomously, the new state of the ingress (and local egress) paths are returned in a single REPT PTHEVT message.

The REPT PTHCHG message contains *pst*, *sst* and *tst* state information

The primary pathState (pst), secondary state (sst) and optional tertiary state (tst) parameters that are returned in the REPT PTHCHG message triggered by a UNI connection request are the actual pst, sst and tst parameters of the connection object after the state change has occurred.

The REPT PTHCHG message contains *pst*, *sst* and *tst* state information

The primary pathState (*pst*), secondary state (*sst*) and optional tertiary state (*tst*) parameters that are returned in the REPT PTHCHG message triggered by a TL1 network connection request (ENT-CONN, DLT-CONN) are the actual *pst*, *sst* and *tst* parameters of the connection object after the state change has occurred.

User defined connection alias returned with REPT-PTHCHG event

The user defined connection alias, if defined, is returned with the event report along with other parameters. If no *connAlias* is defined, then a NULL is returned.

Autonomous path connection configuration change REPT PTHCHG format

Autonomous connection configuration changes that affect the ONNS connection data repository will cause a REPT PTHCHG messages that are output with the CCB= PATH.

Autonomous Path Notification behavior on a protection switch within an M:N protected connection group

In case of a successful protection switch within an M:N protected connection group, the following autonomous notifications are sent with CCB=PATH:

- First of all a REPT PTHEVT notification with code F105 indicating the failure of the working connection, i.e. the working connection and the working path in pathID1 will be in state CON&FAILED
- Then a REPT PTHCHG notification with code C173 for the working connection indicating the completed setup of the protecting path and indicating that the system is waiting for the signal confirmation of the egress port. The connection will be in state CON&SETUP, the protecting path will be associated as pathID2 being in path state CON&SETUP as well. The ID of the associated protecting connection will be given in parameter assocConnID.

- After successful signal confirmation a REPT PTHCHG notifications for the working connection with code C015 indicating that the completed protection switch has been saved in NVM. The connection and the protecting path in pathID2 will both be in state CON&OK&TSTRD
- Additionally a REPT PTHCHG notification for the protecting connection with code C175, also indicating that the completed protection switch has been saved in NVM. The protecting connection will be in state CON&USED and no path will be associated. The ID of the associated working connection will be given in parameter assocConnID.

Autonomous Path Notification behavior on a protection release within an M:N protected connection group

In case of a successful protection release within an M:N protected connection group, the following autonomous notifications are sent with CCB=PATH:

- First of all a REPT PTHEVT notification with code S105 indicating that the original path of working connection is good again and the wait-to-restore time is started. The working connection and the working path in pathID1 will be in state CON&OK&WTORV
- After expiry of the wait-to-restore time a REPT PTHEVT notification with code S102 for the working connection indicating the switch back to the original path. The connection and the working path in pathID1 will be in state CON&OK
- Then a REPT PTHCHG notification for the working connection with code C174 indicating that the protection switch has been released and saved in NVM. The protecting path will not be associated any more in pathID2 and the parameter assocConnID will have the value OPEN.
- Finally a REPT PTHCHG notification for the protecting connection with code C176, also indicating that the protection switch has been released and saved in NVM. The protecting path will be associated again in pathID1 in state RSVD&OK. The connection will be in state CON&STBY and the parameter assocConnID will have the value OPEN.

Output format

Output format for a `REPT PTHCHG` notification message after a connection setup request

The general format of a **`REPT PTHCHG`** notification message from a **`ENT-PATH`** command or a UNI connection setup request is as follows:

```
RTRV-VCGTRIB-STS:UNITE-NE:1-1-#-#-22-v9-all:123456:::dyn=YES;

  LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   ″TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-1::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,\
```

```
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"

   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-2::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"


   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-3::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"


   <listing all VCGTRIB's of this VCG>

   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-12::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"

;
```

Output format for a `REPT PTHCHG` notification message after ED-PATH

The general format of a **`REPT PTHCHG`** notification message from a **`ED-PATH`**, **`ED-PATH-PROT`**, or **`ED-PATH-ROLL`** command is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT PTHCHG
```

```
   ″TIME=ocrTime,DATE=ocrDate:CCB=ED-PATH/ED-PATH-ROLL/ED-PATH-PROT:
    srcNodeIP:srcAid:msgCode,eventText″,
   ″ConnID=connID,ConnStatus=pst&sst&tst,connControl,[pdir],protectionClass,
    bandwidth,[ConnAlias=connAlias],
   [MaxDelay=maxDelay],[SrvID=srvID],[nodeListUsage1=nodeList1],
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[specBlock],[reserved],
   [SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID]
   [,brDestNID=brDestNID,brDestPortID]:
   PathID=pathID1,State=pst&sst&tst,[NodeList=srcNode&nodeIP1&nodeIP2&...&destNode]
   [:PathID=pathID2,State=pst2&sst2&tst2,NodeList=srcNodeIP& ... &destNodeIP]″
;
```

Output format for a **REPT PTHCHG** notification message after **DLT-PATH**

The general format of a **REPT PTHCHG** notification message from a **DLT-PATH** command or a UNI connection delete request is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT PTHCHG
   ″TIME=ocrTime,DATE=ocrDate:CCB=DLT-PATH:srcNodeIP:srcAid:msgCode,eventText″,
   ″ConnID=connID,ConnStatus=pst&sst&tst,connControl,[pdir],[protectionClass],
   bandwidth,[ConnAlias=connAlias],
   [MaxDelay=maxDelay],[SrvID=srvID],[nodeListUsage1=nodeList1],
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[specBlock],[reserved],
   [SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID]
   [,brDestNID=brDestNID,brDestPortID]
   [:PathID=pathID1,State=pst&sst&tst,NodeList=srcNode&nodeIP1&nodeIP2&...&destNode]
   [:PathID=pathID2,State=pst2&sst2&tst2,NodeList=srcNodeIP& ... &destNodeIP]″
;
```

Output format for an autonomous **REPT PTHCHG** notification message

The general format of an autonomous **REPT PTHCHG** notification message is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT PTHCHG
   ″TIME=ocrTime,DATE=ocrDate:CCB=PATH:srcNodeIP:srcAid:msgCode,eventText″,
   ″ConnID=connID,ConnStatus=pst&sst&tst,connControl,[pdir],[protectionClass],
   bandwidth,[ConnAlias=connAlias],
   [MaxDelay=maxDelay],[SrvID=srvID],[nodeListUsage1=nodeList1],
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[specBlock],[reserved],
   [SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID]
   [,brDestNID=brDestNID,brDestPortID]:
   PathID=pathID1,State=pst&sst&tst,[NodeList=srcNode&nodeIP1&nodeIP2&...&destNode]
```

```
   [:PathID=pathID2,State=pst2&sst2&tst2,NodeList=srcNodeIP& ... &destNodeIP]″
;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`rpt-date`** – Reporting date
   This parameter is set to the date when the output message is generated.
   Note: Format uses a 4 digit year instead of a 2 digit year as for host TL1 commands. Compare to **`RTRV-HDR`**.
   Range: YY-MM-DD; where YY=00-99, MM=01-12, DD=01-31.
3. **`rpt-time`** – Reporting time
   This parameter is set to the time when the output message is generated.
   Range: HH:MM:SS; where HH=00-23, MM=00-59, SS=00-59.
4. **`atag`** – Automatic Tag
   This parameter is used for message sequencing. The atag is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Values: 000 (default) – 999
5. **`ocrDate`** – Occurrence Date
   The date parameter is set to the date when the event occurred in the system.
   Values: .YY-MM-DD where YY=00-99, MM=01-12, DD=01-31
6. **`ocrTime`** – Occurrence Time
   The time parameter is set to the time when the event occurred in the system.
   Values: HH-MM-SS where HH=0-23, MM=00-59, SS=00-59.
7. **`comBlock(CB)`**- CCB common block
   This block contains the parts of the command (TL1 or UNI request) causing this REPT PTHCHG notification plus further event info. The following parameters, separated by : will be present:
   - CCB – command code block for the command which caused the event
   - srcNID – srcNID input parameter to the command
   - srcAID – srcAID input parameter to the command

- msgCode – msgCode of the event
- eventText – further event info.

a. **CCB** – Command Code Block

Represents the path command (TL1 or UNI request) causing the change.This confirmation notification indicates an ONNS managed path change was stored in the ONNS Host NVM.

Range: PATH (is specified when the change is not directly caused by path command (TL1 or UNI request), e.g. used for a connection/path change due to restoration. Other values are used when the change is directly caused by a user TL1 or UNI request command: ENT-PATH-UNPROT, ENT-PATH-AUTO, ENT-PATH-N1PLUS1, ED-PATH-PROT, ED-PATH-ROLL, DLT-PATH, ENT-PATH-YCONN, ED-PATH, ENT-PATH-MBYN, ENT-PATH-P1PLUS1NR.

b. **srcNID**

Note: Not present when CCB=DLT-PATH.

ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.

Range: Four dot separated numeric strings. 0 – 255 for each numeric string.

c. **srcAid**

Source node ONNS application access identifier of the ONNS controller OS command interface port.

Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.

d. **msgCode**

Message Code or Status code indicates known or suspected cause or reason for the event, change, or condition.

e. **eventText**

This field records an event description message.

8. **connID**, **connAlias**, **connStatus**, **pdir**, **protectionClass**, **bandwidth**, **maxDelay**, **srvID**, **nodeListUsage1**, **nodeList1**, **nodeListUsage2**, **nodeList2**, **srlgExcludeList**

See the parameter description in the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) section.

9. **connControl** – Connection Control

Indicates whether the ONNS connection is controlled by ONNS or by external control (UNI, eNNI connections).

Type: Value set.

Range:

- ONNS (ONNS connections, created via ENT-PATH)
- EXT (ONNS connections, created via ENT-CONN, UNI or eNNI)

10. **`pdir`** – Path Direction Topology
    Unidirectional A-Z (1WAY) or Bidirectional (2WAY).
    For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
    Type: Value set.
    Range: 1WAY, 2WAY
    Default: 1WAY.
    This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:
    - CCB = **`ENT-PATH-*`**

    Otherwise it is an optional parameter in **`REPT PTHCHG`**.
11. **`protectionClass`**
    This parameter defines the type of restoration protection assigned to the path connection during path setup or after path modification.
    Range:
    - UNPROT (path from source to destination is un-protected)
    - AUTONR (path from source to destination is a non-revertive auto-reroute path connection)
    - N1PLUS1 (paths from source to destination is set up as a 1+1 protected connection)
    - AUTORV (path from source to destination is a revertive auto-reroute path connection)
    - YCONN (paths from the same source to two different destinations are set up as a bridged connection).
    - MBYNWRK (working connection in an M:N connection group)
    - MBYNPRT (protecting connection in an M:N connection group)
    - P1PLUS1NR (paths from source to destination are set up as a 1+1 protected non-revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)
    - P1PLUS1NRRV (paths from source to destination are set up as a 1+1 protected revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)

    This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:
    - CCB = **`ENT-PATH-*`**
    - CCB = **`ED-PATH-PROT`**

    Otherwise it is an optional parameter in **`REPT PTHCHG`**.

12. **`bandwidth`** – Requested bandwidth / signalRate
    Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
    Type: Value set.
    Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
    Default: None.

    This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:

    - CCB = **`ENT-PATH-*`**

    In all other cases, it is an optional parameter in **`REPT PTHCHG`**, that is, it is not mentioned explicitly as mandatory.

13. **`maxDelay`** – Maximum transport delay
    Maximum transport delay through all nodes in the path. Total path delay must not exceed this value.
    Range: 0, is used to select no delay limit; 1 – 32767, maximum transport delay in milliseconds.
    Default: 0.

    This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:

    - CCB = **`ENT-PATH-*`**
    - CCB = **`ED-PATH-PROT`**
    - CCB = **`ED-PATH-ROLL`**

    Otherwise it is an optional parameter in **`REPT PTHCHG`**.

14. **`srvID`** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    Type: Integer.
    Range: 1 – 20000000
    Only one *srvID* is accepted. The value "ALL" is not supported.

15. **`nodeListUsage1`**
    For connection setup NN used/uses the list of nodes included in the *nodeList1* array block in a manner specified by the *nodeListUsage1* variable. Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command. An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent *nodeID* will be ignored in EXCLUDE *nodeList1* during path setup. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
    Type: Value set.

Range:

- EXPLICIT, *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.)
- EXCLUDE, nodes in the *nodeList1* must all be excluded from the path created. In the case when the *protectionClass* is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the *nodeList1* below also apply to the second or restoration path.

This is a mandatory output parameter in **REPT PTHCHG** in the following cases:

- CCB = **ENT-PATH-*** and parameter exists
- CCB = **ED-PATH-*** and parameter exists

Otherwise it is an optional parameter in **REPT PTHCHG**.

16. **nodeList1**

When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidx]).

When *nodeListUsage1* = EXCLUDE: List of nodes that must all be excluded from both paths. The *nodeID* values are shown in the from: nid1([&nid2& …&nidx]).

If the preceding required *nodeListUsage1* field is not supplied, this field must not be present. If user specifies empty EXCLUDE *nodeList1*, the connection should use indirect path from source node to destination node (that is, at least one intermediate node is needed). If user specifies empty EXPLICIT *nodeList1*, the connection should use a direct path from source to destination node (that is, no intermediate node is allowed).

Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.

Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

Range: nid1([&nidx])* or empty list (where X <= 25 when *nodeListUsage1* = EXPLICIT) (where X <= 25 when *nodeListUsage1* = EXCLUSIVE)

This is a mandatory output parameter in **REPT PTHCHG** in the following cases:

- CCB = **ENT-PATH-*** and parameter exists
- CCB = **ED-PATH-*** and parameter exists

17. **nodeListUsage2**

For connection setup NN used/uses the list of nodes included in the *nodeList2* array block in a manner specified by the *nodeListUsage2* variable. Both, *nodeListUsage2* and *nodeList2*, fields belong together and is reported when supplied as input to the **ENT-PATH** command. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.

This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.

This parameter shall also be allowed when *protectionClass* is P1PLUS1NR.
Type: Value set.
Range: EXPLICIT, *nodeList2* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.).

This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:

- CCB = **`ENT-PATH-*`** and parameter exists
- CCB = **`ED-PATH-*`** and parameter exists

18. **`nodeList2`**
When *nodeListUsage2* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([[&nid2& …&nidx]]).
If the preceding required *nodeListUsage2* field is not supplied, this field must not be present. If user specifies empty EXPLICIT *nodeList2*, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
This parameter shall also be allowed when *protectionClass* is P1PLUS1NR
Type: &-separated list of *nodeIDs*. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
Range: nid1([&nidX])* or empty list where X <= 8 when *nodeListUsage1* = EXPLICIT.

This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:

- CCB = **`ENT-PATH-*`** and parameter exists
- CCB = **`ED-PATH-*`** and parameter exists

19. **`srlgExcludeList`** – SRLG Exclude List
List of Shared Risk Link Group identifiers that are to be used to exclude links from the connection path(s) being created.
Type: &-separated list of numeric strings | empty | 0
Example: 33&49&64
Range: up to 25 numeric strings, each in the range 1 – 65535. The value 0 (default) or an empty field indicates no constraint are applied.

This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:

- CCB = **`ENT-PATH-*`**
- CCB = **`ED-PATH-*`**
- CCB = **`ED-ROLL`**

20. **`linkInclude`** – VPN Identifier
    A connection constraint link property such as *linkIncludeID* that must be valid to include the link in the connection path(s) to be created. *linkIncludeID* is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
    Range: 1 – 65535. 0 (default) or an empty field indicates no VPN constraint is applied.
    Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.
21. **`srcNID`** – NN source node Network IP Access identifier (nid).
    ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
    Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
22. **`srcPortID`**
    ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
    Value: dash-separated list of 5 numeric strings
    Example: 1-1-3-1-4
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
23. **`destNID`** – Destination nodeID
    ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
    Type: Four dot separated numeric strings. 0 – 255 for each numeric string.
24. **`destPortID`**
    ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
    Value: dash-separated list of 5 numeric strings
    Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
25. **`brDestNID`** – Bridged Destination NodeID
    ONNS bridged destination nodeID. Together with the following brDestPortID field defines the second associated Egress connPoint that connects the destination node to the edge/gateway switch or client router in case of protectionClass YCONN.
    Type: Four dot separated numeric strings. 0 – 255 for each numeric string.

26. **`brDestPortID`** – Bridged Destination PortID
ONNS bridged destination node egress *portID* in case *protectionClass* = YCONN. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the brDestNID egress connPoint. 4.1; 5; 6
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).

27. **`pathID1`** – Path ID 1
The PathID of the 1st path that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255); (Index: Numeric string 0-65535).
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111

28. **`pathState1`** – Path State 1
Path State1 = *pst*& *sst*[*&tst*]. pathState1 represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.

PathState1 is composed out of

- *pst* (PrimaryState)
- *sst* (SecondaryState)
- *tst* (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets.

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values:

- OK: Conn good
- FAILED: PATH failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed

- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

Range *pst* values:

- RSVD: Reserved, for paths associated to protecting connections within an M:N protection group, while not protecting a working traffic.

29. **`pNodeList1`** – Path Node List 1
The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

30. **`pathID2`** – Path ID 2
The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255), (Index: Numeric string 0-65535).
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111

31. **`pathState2`** – Path State 2
pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of:

- *pst* (PrimaryState)
- *sst* (SecondaryState)
- *tst* (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand ("&"). Individual status values are 7 characters or less.
Type: up to three &-separated value sets.
Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values:

- OK: Conn good
- FAILED: PATH failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

32. **`pNodeList2`** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

33. **`reqType`**
Note: Only present if CCB=ED-PATH[-mod].
This parameter indicates the operation that is to be performed on the connection specified.
Type: Alphanumeric value set
Range:
    - SETPARAM: modifies the value of *rstrPriority*, *wtr*, *htmr*, or *switchBack* of the connection to the value specified in the spec block
    - SETPROT: converts the connection specified by its *connID* to a new protection type specified by the *connMode* parameter in the spec block.

    Default: None.This are a mandatory output parameter in REPT PTHCHG in the following cases: CCB=ED-PATH-* It is not sent in REPT PTHCHG otherwise.
This is a mandatory output parameter in **`REPT PTHCHG`** in the following cases:
    - CCB = **`ED-PATH-*`**

    Otherwise it is not sent in **`REPT PTHCHG`**.

34. **`msgCode`**
Message Code or Status code indicates a known result.
Type: Alphanumeric string of length 4

Spec block parameters

1. **`connMode`** – Connection Mode
The connection mode determines ONNS treatment of the failed path after an auto-reroute restoration.
Type: Alphanumeric value set.
Range:
    - UNPROT: converts a auto connection to a unprotected
    - N1PLUS1: Network 1+1 protected connection
    - NONREVERTIVE: Auto-reroute non-revertive
    - EVERTIVE: Auto-reroute revertive
    - YCONN: Y-connection
    - BRIDGEY: Y connection Protection

- MBYNWRK: Working connection within an M:N protection group
- MBYNPRT: Protecting connection within an M:N protection group.

2. **`rstrPriority`** – Restoration Priority
   This parameter describes the restoration priority of the connection.
   Range: 0 (default) – 7. Lower values represent higher priority.
3. **`connectionPriority`** – Connection Priority
   Connection priority allows connection setup of a higher priority connection to utilize the existing path through the network when the existing connection is pre-emptible.
   Range: 0 (default) – 7. 0 = locked, 1 – 7 = Priority, Connection may be pre-empted.
4. **`disjointness`**
   The actual level of disjointness of the calculated and established connection. The parameter is not filled when protectionClass is UNPROT.
   Type: Alphanumeric value set.
   Range:
   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.

   Default:
   This are a mandatory output parameter in REPT PTHCHG in the following cases:
   - CCB=ENT-PATH-* and parameter exits
   - CCB=ED-PATH-* and parameter exits
   - MAXIMAL: maximal node disjoint path.

   It is an optional parameter in REPT PTHCHG otherwise.
5. **`connGrpID`**
   Connection Group ID for an existing ONNS connection to report the connection group the connection is member of or ″UNDEFINED″, if connection is not member of a group.
   For connection groups identified by A VCG EDGE port, *connGrpID* is formed from the *SrcNodeID* and the Ingress Port #, where 1675 LambdaUnite MSS uses bay =1= shelf. The connection group ID is automatically assigned to a connection, when it is setup. The connection group ID will consist of the nodeID and the EDGE port AID in case the EDGE port is a VCG. The connection group ID will be ″UNDEFINED″ otherwise. For M:N protected connection groups, *connGrpID* is formed from the *SrcNodeID* and a group index chosen by the source node to be unique within this node. The connection group ID is automatically assigned to a connection, when it is setup. The connection group ID will consist of the node ID and a group index in case the connection is member of an M:N connection group. The connection group ID will be ″UNDEFINED″ otherwise.
   Type: Character string of length up to 25.

Range:

- UNDEFINED – special value for connections not being member of a group
- nodeID&Port AID – value for connections being member of a ″GbE Connection Group″
- nodeID&groupIndex – value for connections being member of a ″M:N Connection Group″
- nodeID: four dot separated numeric strings: each string 0..255
- Port AID – four dot separated numeric strings: representing Bay/Frame #, Shelf #, Slot/Pack #, Port # with: Bay/Frame #1, Shelf #1, Slot/Pack #1..39, Port #1..8)
- groupIndex – integer value specifying an M:N connection group 0-65535.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.

6. **`protRatio`** – Protection ratio
   For each working or protecting connection of an M:N protected connection group this parameter will report the concrete protection ratio as specified on creation of the group. For connections not being member of an M:N protected connection group, this parameter is not reported at all.
   Type: Alphanumeric value set
   Range. 1BY1 (ratio of 1 to 1 in number of protecting and working connections), 1BY2 (ratio of 1 to 2 in number of protecting and working connections), 1BY3 (ratio of 1 to 3 in number of protecting and working connections).
7. **`assocConnID`** – Associated Connection ID
   Associated Connection ID for an existing ONNS connection of an M:N connection group to report the connection ID of the associated connection within the group, when a protection switch is active, i.e. for a working connection the ID of the associated protecting connection and for a protecting connection the ID of the associated working connection is reported. The associated connection ID will be ″OPEN″ for connections of an M:N connection group not involved in a protection switch. The associated connection ID will not be reported for connections not being member of an M:N connection group.
   Connection ID for an existing ONNS path connection service. Reports only the connections originating at the specified source node. ConnID is formed from the SrcNodeID, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.
   Type: Character string of length up to 39.
   Range:

ALL - special value to retrieve all connection records | nodeID&Tributary AID&Index - to retrieve a specific entry, with:

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with: Bay/Frame #1, Shelf #1, Slot/Pack # 1..39 , Port #1..16, Tributary #1..192.
- Index: Numeric string 0-4294967295

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x. Example: 29.13.196.203&1-1-13-15-1&271584

nodeID&Tributary AID range - if port # specifies a VCG to retrieve a connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with: Bay/Frame #1, Shelf #1, Slot/Pack # 1..39 , Port #1..9, Tributary #ALL – special value to retrieve all connection records associated with the port ID.

As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG. The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x. Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

nodeID&defaultTributary AID&Index - to retrieve a protecting connection of an M:N connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- defaultTributary AID: five dash separated numeric strings with Bay/Frame #1, Shelf #1, Slot/Pack ##, Port ##, Tributary ##
- Index: Numeric string 0-4294967295

  The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.

nodeID&groupIndex -to retrieve all connections belonging to an M:N protected connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- groupIndex: integer value specifying a connection group within nodeID 0-65535.

The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Default: n/a.

This are a mandatory output parameter in REPT PTHCHG in the following cases:

- CCB=ENT-PATH-* and parameter exists
- CCB=PATH and parameter exists

It is an optional parameter in REPT PTHCHG otherwise.

8. **`htmr`** – Hold-Off Timer
   This parameter applies to 1+1 protected and permanent 1+1 protected connections.
   This parameter specifies the value of the hold-off timer for 1+1 protection switching.
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms]
   Range:

   For high order path protection groups, the range is:

   - 0 .. 99 ms in 1 ms steps and
   - 100 .. 10000 ms in100 ms steps

   For low order path protection groups, the range is:

   - 0, 25, 50, 75 ms (thus in steps of 25 ms) and
   - 100 .. 10000 ms in 100 ms steps.

   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block.
   Default: 0
   This parameter shall relate to the *htmr* parameter provisioned via the TL1 commands **`ENT-PATH`** or **`ED-PATH-PROT`**. Configuration of path protection group parameter via **`ED-PROTN-GRP`** is not forbidden, but should be discouraged, since this configuration is not reflected in the ONNS view. During normal operation of an ONNS connection, the **`ED-PROTN-GRP`** parameters are considered but during ONNS protection conversion the ONNS stored path protection group parameters are considered.

   This are a mandatory output parameter in REPT PTHCHG in the following cases:

   - CCB=ENT-PATH-* and parameter exists
   - CCB=PATH and parameter exists

9. **`wtr`** – Wait To Restore Time
   This parameter specifies the wait-to-restore time for revertive 1+1 protection switching.
   Parameter does not apply to a 1+1 protected connection revertive.
   Parameter applies to a permanent 1+1 protected connection with revertive SNCP/UPSR protection.
   Type: numeric string [minute].
   Range: 0 – 60.
   Default:
   None.
   The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block. Default Value: 0
   This parameter shall relate to the *wtr* parameter provisioned via the TL1 commands **`ENT-PATH`** or **`ED-PATH-PROT`**. Configuration of path protection group parameter via

ED-PROTN-GRP is not forbidden, but should be discouraged, since this configuration is not reflected in the ONNS view. During normal operation of an ONNS connection, the **ED-PROTN-GRP** parameters are considered but during ONNS protection conversion the ONNS stored path protection group parameters are considered.

This are a mandatory output parameter in REPT PTHCHG in the following cases:

- CCB=ENT-PATH-* and parameter exists
- CCB=PATH and parameter exists

Example output

The following notification examples detail specific situations and the corresponding message codes used in the REPT PTHCHG notification to alert the OS system of the occurrence.

New Connection Saved

- C001 – New UNPROT Connection Saved in NVM
- C002 – New AUTONR Connection Saved in NVM
- C003 – New AUTORV Connection Saved in NVM
- C004 – New N1PLUS1 Connection Saved in NVM
- C036 – New YCONN Connection Saved in NVM
- C051 – New MBYNWRK Connection Saved in NVM
- C052 – New MBYNPRT Connection Saved in NVM
- C061 – New P1PLUS1 Connection Saved in NVM

REPT PTHCHG Output example for ENT-PATH-UNPROT

The following is an example of the REPT PTHCHG message for a input TL1 ENT-PATH-UNPROT command that has been saved in the ONNS NVM connection data repository:

```
   LT-ONNS101 2002-07-01 08:00:00
A  01135 REPT PTHCHG
   ”TIME=ocrTime,DATE=ocrDate:CCB=ENT-PATH-UNPROT:125.114.48.12:SNN:C001:\”New
      UNPROT Connection Saved in NN NVM\”,”
   ”ConnID=125.14.48.2&1-1-3-4-1&1,ConnStatus=CON&OK,ONNS,2WAY,UNPROT,STS48,
   connAlias=AEXP-NY-BOS-ckt,
   MaxDelay=65,SrvID=135.2.5.4-
      37,Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,SrlgExclude=59&630&1943,IncludeID=
   SrcNID=125.14.48.2,1-1-3-4-1&1-1-3-4-1,DestNID=125.9.43.6,1-2-1-2-1&1-2-1-2-1:
   PathID=125.14.48.2-28123,State=CON&OK,
   NodeList=125.14.48.2&nodeID1&nodeID2&...&78.22.3.17&125.19.43.6”
;
```

Modify Connections

- C005 – Edited UNPROT Connection Saved in NVM
- C006 – Edited AUTONR Connection Saved in NVM
- C007 – Edited AUTORV Connection Saved in NVM
- C008 – Edited N1PLUS1 Connection Saved in NVM
- C037 – Edited YCONN Connection Saved in NVM
- C017 – Edited MBYNWRK Connection Saved in NVM
- C018 – Edited MBYNPRT Connection Saved in NVM
- C062 – Edited P1PLUS1 Connection Saved in NVM
- C009 – Edited Connection Modified (Not Saved in NVM)

REPT PTHCHG output example for ED-PATH-PROT and ED-PATH-ROLL

The following is an example of the REPT PTHCHG message for a input TL1 ED-PATH-PROT or ED-PATH-ROLL change that has been saved in the ONNS NVM connection data repository:

```
   LT-ONNS101 2002-07-01 08:00:00
A  0013495 REPT PTHCHG
   ″TIME=ocrTime,DATE=ocrDate:CCB=ED-PATH-PROT:125.114.48.12:SNN:C008:\″Edited
      N1PLUS1 Connection Saved in NVM \″,″
   ″ConnID=125.114.48.12&1-1-23-4-13&1,ConnStatus=CON&OK,ONNS,1WAY,N1PLUS1,STS12,
   connAlias=AEXP-NY-BOS-ckt,
   MaxDelay=99,SrvID=125.4.3.675,
   Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,
   SrlgExclude=401&319&1024,IncludeID=0,disjoint=COMPLETE,,
   SrcNID=125.114.48.12,1-1-23-4-13,DestNID=125.19.43.6,1-1-12-2-25&1-2-12-2-25:
   PathID=125.114.48.12-12123,State=CON&OK,
   NodeList=125.114.48.12&nodeID1&nodeID2&...&125.19.43.6:
   PathID=125.114.48.12-11013,State=CON&OK,
   NodeList=125.114.48.12&nodeID4&nodeID5&...&125.19.43.6″
;
```

Connection and Path Removed

- C010 – Unprotected Connection Removed from NVM
- C011 – AUTONR Connection Removed from NVM
- C012 – AUTORV Connection Removed from NVM
- C013 – N1PLUS1 Connection Removed from NVM
- C014 – N1PLUS1 Path Removed from connection and NVM
- C019 – A Path was deleted from the connection and NVM
- C038 – YCONN Connection Removed from NVM
- C039 – YCONN Path Removed from connection and NVM

- C053 – MBYNWRK Connection Removed from NVM
- C054 – MBYNWRK Path Removed from connection and NVM
- C063 – P1PLUS1 Connection Removed from NVM
- C064 – P1PLUS1 Path Removed from connection and NVM
- C055 – MBYNPRT Connection Removed from NVM
- C066 – Connection Deletion failed due to signalling problems

This code is generic and applies to all ONNS connection types.

REPT PTHCHG Output example for the DLT-PATH or ED-PATH command

The following is an example of the REPT PTHCHG message for a TL1 DLT-PATH or ED-PATH command that has been saved in the ONNS NVM connection data repository:

```
   LT-ONNS111 2002-07-01 08:00:45
A  002478 REPT PTHCHG
   ″TIME=ocrTime,DATE=ocrDate:CCB=DLT-PATH::SNN:C013:\″N1PLUS1 Connection Removed
      from NN NVM\″,″
   ″ConnID=125.114.48.12&1-1-3-4-1&1,ConnStatus=DLT,ONNS,1WAY,N1PLUS1,STS12,
      connAlias=AEXP-NY-BOS-ckt,
   MaxDelay=99,SrvID=135.2.5.4-
      39,Exclude=135.13.42.3&135.213.242.103&135.13.198.22,,SrlgExclude=401&319&1024,IncludeID=
   SrcNID=125.114.48.12,1-1-23-4-13,DestNID=125.19.43.6,1-1-12-2-25:
   PathID=125.114.48.12-29123,State=DLT:
   PathID=125.114.48.12-29128,State=DLT″
;
```

Connection Restorations

- C015 – Connection restored and Saved in NVM applies to AUTONR and AUTORV connections
- C020 – Failed Path Deleted and Removed from NVM
- C025 – AUTORV Connection restored and Saved in NVM applies if the AUTORV is switched back to original path
- C200 – \″ Connection Setup completed and Waiting for signal confirmation of egress port \″, with connStatus = CON&SETUP
- C201 – \″ Restoration Setup completed and Waiting for signal confirmation of egress port \″, with connStatus = CON&SETUP
- C210 – \″ Connection Setup completed and No signal at ingress port \″, with connStatus = CON&NO-IN
- C211 – \″ Restoration Setup completed and No signal at ingress port \″, with connStatus = CON&NO-IN

REPT PTHCHG output example for PATH restoration for AUTONR

The following is an example of the REPT PTHCHG message for a path restoration change that has been saved in the ONNS NVM connection data repository. An TL1 initiated auto-rerouting connection has failed and has been restored. This REPT PTHCHG is sent after the restoration result is in NVM, and while the failed path is being torn down:

```
   LT-ONNS101 2002-07-01 08:00:00
A  00134 REPT PTHCHG
   ″TIME=ocrTime,DATE=ocrDate:CCB=PATH:125.114.48.12:SNN:C015:\″Connection restored
      and Saved in NVM\″,″
   ″ConnID=125.114.48.12&1-1-3-4-1&1,ConnStatus=CON&OK,ONNS,2WAY,AUTONR,STS192,
   connAlias=AEXP-NY-BOS-ckt,
   MaxDelay=199,,,SrlgExclude=630&820&720,IncludeID=0,RstrPriority=3,,
   SrcNID=125.114.48.12,1-1-3-4-1,DestNID=125.6.19.43,1-1-14-2-1:
   PathID=125.114.48.12-12123,State=CON&OK,
   NodeList=125.114.48.12&nodeID1&nodeID2&...&125.6.19.43:
   PathID=125.114.48.12-11013,State=CON&TRDOWN,
   NodeList=125.114.48.12&nodeID4&nodeID5&...&125.6.19.43″
;
```

Certain Failures at connection creation

- C250 – \″ Connection Setup failed
  Note: All *errorCodes* and ″*errorMessages*″ from ENT-PATH path setup error responses can occur in this situation.
- C251 – \″ Restoration Setup failed
  Note: All *errorCodes* and ″*errorMessages*″ from ENT-PATH path setup error responses can occur in this situation.

Edit of Connections

- C170 – N1PLUS1 Path Added to Connection and Saved in NVM
- C065 – P1PLUS1 Path Added to Connection and Saved in NVM
- C171 – Connection Bridged and ready for roll, Saved in NVM
- C172 – Connection Rolled to new path and Saved in NVM
- C173 – Working connection, M:N protection established, protecting path allocated successfully, waiting for confirmation, \″PathID=*PathID*″
- C174 – Working connection, M:N protection released, protecting path returned successfully, \″PathID=*PathID*″
- C175 – Protecting connection, M:N protection established, protecting path provided successfully to working connection, \″PathID=*PathID*″
- C176 – Protecting connection, M:N protection released, protecting path reallocated successfully, \″PathID=*PathID*″
- C177 – Working connection, M:N protection switch finished successfully and Saved in NVM

- C178 – Protecting connection, M:N protection switch finished successfully and Saved in NVM
- C179 – Working connection, M:N protection released, Saved in NVM
- C180 – Protecting connection, M:N protection released, Saved in NVM

Related TL1 messages

- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "REPT-PTHEVT notification" (p. 3-2175)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "REPT-NNCHG notification" (p. 3-2137)
- "REPT-NNEVT notification" (p. 3-2144)
- "RTRV-LOG-NTFCN: Retrieve notification log" (p. 3-2464)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "ED-NNCNFG: Modify ONNS system configuration" (p. 3-450)

REPT-PTHEVT notification

Overview

The **REPT PTHEVT** notification is used to notify about autonomously triggered changes of the status of ONNS connections. REPT PTHEVT notifications are sent also for ONNS connections requested by network connections (by UNI, eNNI, or ENT-CONN).

REPT PTHEVT notification messages are generated autonomously by ONNS to report network connection status changes, and path state change events.

General description of ONNS events

ONNS reports events to NMS when certain network connection state changes occur or path state events that relate to connections originating from this node.

In general, destination nodes report client port alarms as a path fault event via the signaling network to the controlling source node. The source node will verify the fault state by checking the ingress signal states. Then ONNS shall send up a REPT PTHEVT: failed_path notification message indicating that the optical path has failed, and then ONNS begins the rerouting process to create a new optical path if one is required by the provisioned restoration service.

ONNS reports events to the Network Management users that are provisioned to receive the ONNS PTHCHG's. Message provisioning occurs via the Host NE's ED-USER-SECU command.

When the source node has verified a path failure, it shall send out a REPT PTHEVT notification indicating that the path has failed. When the restoration path switch has occurred successfully, the source node shall report the REPT PTHCHG: restoration message indicating that the restoration path is operational and reports its pathID. When the connection status changes to CON&OK, ONNS shall report the event by a REPT PTHEVT: connection_OK notification. When the ingress signal state of source or destination port is determined to have failed and restoration would not be successful, ONNS shall not proceed with restoration and ONNS report a REPT PTHEVT: Input_failed notification.

1+1 protection switching events are not reported via a REPT PTHEVT notification

Protection switching messages is not provided to the MS using the REPT PTHEVT TL1 message. By definition, a protection switching event occurs when the service switches from working to protection or active to standby and vice versa.

Unique autonomous REPT PTHEVT notification messages

REPT PTHEVT shall not report any autonomous notification messages that are reported by other ONNS REPT notifications. REPT PTHEVT shall report connection status changes and path state changes. As an example, REPT PTHCHG autonomous notification messages report when connection configuration or provisioning changes are made. REPT PTHEVT does not report the connection provisioning changes.

User defined connection alias returned with REPT-PTHEVT

The user defined connection alias, if defined, is returned with the event report along with other parameters. If no connAlias is defined, then a NULL is returned.

Output format

The general format of a **REPT PTHEVT** autonomous notification message is as follows:

```
   srcTid rpt-date rpt-time
A  atag REPT PTHEVT
   ″TIME=ocrTime,DATE=ocrDate:srcNID:srcAid:msgCode::condType:condDescr:\″
      eventText\″:[eventData]″
   ″ConnID=connID,ConnStatus=pst&sst[&tst],connControl,
   ConnAlias=connAlias,
   PathID=pathID,State=pst&sst[&tst]″
   [″errorMsgCode:\″ ErrorText String \″:[errorData]″]
;
```

Output parameters

Command parameters

1. **srcTid** – Source Node Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore “_” and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **rpt-date** – Reporting date
   This parameter is set to the date when the output message is generated.
   Note: Format uses a 4 digit year instead of a 2 digit year as for host TL1 commands. Compare to RTRV-HDR.
   Range: YY-MM-DD; where YY=00-99, MM=01-12, DD=01-31.
3. **rpt-time** – Reporting time
   This parameter is set to the time when the output message is generated.
   Range: HH:MM:SS; where HH=00-23, MM=00-59, SS=00-59.

4. **`atag`** – Automatic Tag
   This parameter is used for message sequencing. The atag is available per user session. The number is incremented by one for each autonomous message, and the messages are transmitted to the user in that same order.
   Values: 000 (default) – 999
5. **`ocrDate`** – Occurrence Date
   The date parameter is set to the date when the event occurred in the system.
   Values: .YY-MM-DD where YY=2000-2099, MM=01-12, DD=01-31
6. **`ocrTime`** – Occurrence Time
   The time parameter is set to the time when the event occurred in the system.
   Values: HH-MM-SS where HH=0-23, MM=00-59, SS=00-59.
7. **`comBlock (CB)`**- CCB common block
   This common block contains the message-dependent parameters.
   a. **`srcNID`**
      ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
      Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
   b. **`srcAid`**
      Source node ONNS application access identifier of the ONNS controller OS command interface port.
      Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
   c. **`msgCode`**
      Message Code or Status code indicates known or suspected cause or reason for the event, change, or condition.
   d. **`condType`** – Condition Type
      This is a short hand notation for the *condDescr*.
      Range: CLR (Path condition cleared), PFL (Path Failure), LOS (Loss of Signal), PEF (Port Equipment Failure), PEV (Generic Path event).
   e. **`condDescr`** – Condition Description
      This is a more detailed description of the condition and consists of the following comma separated fields: *category*, *entity*, *protType* (may be empty), *description*, *additional text* (optional, if omitted, the comma before might be omitted, too).
      Range for *category*: TRAN (Transmission), RESTR (Restoration), PRTST (Port_state).
      Range for *entity*: PATH (path connection), CONN (End to end connection), XCON (NE cross-connection), IPORT (Ingress port), GRP (connection group).

Range for *protType*: UNPROT (Unprotected path), N1PLUS1 (Netwrk1+1 protected path), AUTONR (Auto-rerouted protection path, non-revertive), AUTORV (Auto-rerouted protection path, revertive), YCONN (Y-Connection), MBYNWRK (Working connection of M:N protection group), P1PLUS1 (Permanent 1+1 connection).

Range for *description*: FAILED (path failure), FAULT (path fault), CLEAR (path problem cleared), RSTRFL (connection restore failed), WARNING (path warning), PRTFL (Protection Switch Failed).

f. **eventText**

This field records an event description message.

g. **eventData**

Further event data as named-parameter=value pair(s). Comma separated when multiple pairs.

8. **connID**, **connAlias**, **connStatus**, **pathID1**, **pathState1**, **pathID2**, **pathState2**
See the parameter description in the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) section.

9. **connControl** – Connection Control
Indicates whether the ONNS connection is controlled by ONNS or by external control (UNI, eNNI connections).
Type: Value set.
Range:
   - ONNS (ONNS connections, created via ENT-PATH)
   - EXT (ONNS connections, created via ENT-CONN, UNI or eNNI)

10. **errorMsgCode** – Error Message Code
Message Code or Status code indicates known or suspected cause or reason for the event, change, or condition.

11. **errorText**
Additional free format error message text

12. **errorData**
Further error data which reports named-parameter=value pair(s) for the paths.

Example output

Declaration of a PATH failure event

The following REPT PTHEVT autonomous notification message illustrates how an unprotected path connection fault is reported. The REPT PTHEVT autonomous message declares the occurrence of a path failure on *pathID* of connection *connID*, and describes the state of the path:

```
   WS-LT-ONNS-M 2002-08-01 08:00:00
A  00313 REPT PTHEVT
   ″TIME=07-59-59,DATE=2002-08-01:135.13.16.45:SNN:F102::PFL:RESTR,CONN,
AUTONR,FAILED:\″Connection Failed, Restoration In Progress\″:″
   ″ConnID=135.13.196.45&1-1-4-2-1&1,ConnStatus=CON&FAILED,ONNS,
```

```
ConnAlias=AEXP-NY-BOS-ckt,PathID=135.13.196.45-987654,State=CON&FAILED"
;
```

- F101 – PFL:TRAN,CONN,UNPROT,FAILED – Unprotected Connection Failed
- F108 – LOS:TRAN,CONN,UNPROT,FAULT – Incoming signal fault, LOS fault at Ingress Node
- F116 – PEF:PRTST,IPORT,UNPROT,FAILED – EDGE Port failure, UNPROT Connection Path failure
- F210 – PEF:PRTST, IPORT UNPROT, FAILED – UNI Port failure, UNPROT Connection failure
- F220 – PEF:PRTST, IPORT UNPROT, FAILED – eNNI Port failure, UNPROT Connection failure
- F102 – PFL:RESTR,CONN,AUTONR,FAILED – Connection Failed, Restoration In Progress
- F106 – PFL:RESTR,CONN,AUTONR,RSTRFL – Restoration Failed, Connection Failed
- F117 – PEF:PRTST,IPORT,AUTONR,FAILED – EDGE Port failure, no restoration attempted
- F211 – PEF:PRTST, IPORT, AUTONR, FAILED – UNI Port failure, no restoration attempted
- F221 – PEF:PRTST, IPORT, AUTONR, FAILED – eNNI Port failure, no restoration attempted
  No restoration attempt (non-revertive) is initiated by the source node because fault is detected outside of the ONNS domain.
- F109 – LOS:TRAN, CONN, AUTONR, FAULT – Incoming signal fault (LOS), no restoration attempted.
  Restoration attempt is initiated by the source node; destination node detects that the fault is outside of the ONNS domain and rejects the restoration attempt
- F113 – LOS:TRAN, CONN, AUTONR, FAULT – Incoming signal fault, DestNode, restoration attempt aborted
- F105 – PFL:RESTR, CONN, AUTORV, FAILED – Connection Failed, Restoration In Progress
- F107 – PFL:RESTR, CONN, AUTORV, RSTRFL – Restoration Failed, Connection Failed
- F118 – PEF:PRTST, IPORT AUTORV, FAILED – EDGE Port failure, no restoration attempted
- F213 – PEF:PRTST, IPORT AUTORV, FAILED – UNI Port failure, no restoration attempted

- F223 – PEF:PRTST, IPORT AUTORV, FAILED – eNNI Port failure, no restoration attempted
  No restoration attempt (revertive) is initiated by the source node because fault is detected outside of the ONNS domain.
- F110 – LOS:TRAN, CONN, AUTORV, FAULT – Incoming signal fault (LOS), no restoration attempted
  Restoration attempt is initiated by the source node; destination node detects that the fault is outside of the ONNS domain and rejects the restoration attempt.
- F114 – LOS:TRAN, CONN, AUTORV, FAULT – Incoming signal fault, DestNode, restoration attempt aborted
- F103 – PFL:TRAN,CONN,N1PLUS1,FAILED – N1PLUS1 Connection Failed
- F104 – PFL:TRAN,PATH,N1PLUS1,FAULT – N1PLUS1 Path Failed
- F111 – LOS:TRAN,CONN,N1PLUS1,FAULT – Incoming signal fault, LOS fault at Ingress Node
- F119 – PEF:PRTST,IPORT,N1PLUS1,FAILED – EDGE Port failure, N1PLUS1 Connection Failure
- F212 – PEF:PRTST, IPORT, N1PLUS1, FAILED – UNI Port failure, N1PLUS1 Connection Failure
- F224 – PEF:PRTST, IPORT, N1PLUS1, FAILED – eNNI Port failure, N1PLUS1 Connection Failure
- F214 – PFL:TRAN,CONN,P1PLUS1,FAILED – P1PLUS1 Connection Failed
- F215 – PFL:TRAN,CONN,P1PLUS1,FAILED – P1PLUS1 Connection Failed, Restoration In Progress
- F216 – PFL:TRAN,CONN,P1PLUS1,RSTRFL – P1PLUS1 Connection Failed, Restoration Failed
- F217 – PFL:TRAN,PATH,P1PLUS1,FAULT – P1PLUS1 Path Failed
- F218 – PFL:TRAN,PATH,P1PLUS1,FAULT – P1PLUS1 Path Failed, Restoration In Progress
- F219 – PFL:TRAN,PATH,P1PLUS1,RSTRFL – P1PLUS1 Path Failed, Restoration Failed
- F135 – PFL:TRAN, CONN, YCONN, FAILED – YCONN Connection Failed
- F136 – PFL:TRAN, PATH, YCONN, FAULT – YCONN Path Failed
- F137 – LOS:TRAN, CONN, YCONN, FAULT – Incoming signal fault, LOS fault at Ingress Node
- F138 – LOS:TRAN, CONN, YCONN, FAULT – Incoming signal fault, LOS fault at DestNode
- F139 – PEF:PRTST, IPORT, YCONN, FAILED – EDGE Port failure, YCONN Connection Failure
- F120 – PFL:PROTN, CONN, MBYNWRK, FAILED – Connection Failed , Protection Switch In Progress

- F121 – PFL:PROTN, CONN, MBYNWRK, PRTNP – Protection Switch Not Possible, No MBYNPRT Connection Available, Connection Failed
- F122 – LOS:TRAN, CONN, MBYNWRK, FAULT – Incoming signal fault (LOS), no protection switch attempted
- F123 – LOS:TRAN, CONN, MBYNWRK, FAULT – Incoming signal fault, DestNode, protection switch attempt aborted
- F124 – PEF:PRTST, IPORT, MBYNWRK, FAILED – EDGE Port failure, no protection switch attempted
- F126 – PFL:TRAN, PATH, MBYNWRK, FAULT – Standby Path Failed
- F127 – PFL:PROTN, CONN, MBYNWRK, PRTFL – Protection Switch Failed, Connection Failed
- F125 – PFL:TRAN, CONN, MBYNPRT, FAILED – MbyN Protecting Connection Failed

Clearing of a REPT PTHEVT event or restoration failure event

The following REPT PTHEVT autonomous message reports that the LOS cleared. If the connection has two paths, the *pathID* and *pathState* for both paths is included:

```
   WS-LT-ONNS-A 2002-07-01 08:11:05
A  01104 REPT PTHEVT
   ″TIME=07-59-59,DATE=2002-08-
      01:135.13.19.45:SNN:S101::CLR:TRAN,CONN,UNPROT,CLEAR:\″ Unprotected path is
      OK\″:″
   ″ConnID=135.13.196.45&1-1-4-2-1&1,ConnStatus=CON&OK,EXT,
   ConnAlias=AEXP-NY-BOS-ckt,
   PathID=135.13.196.45-987654,State=CON&OK″
;
```

- S101 – CLR:TRAN, CONN, UNPROT, CLEAR – Unprotected Path is OK
- S106 – CLR:TRAN, IPORT, UNPROT, CLEAR – EDGE Port, IN Signal Fault has cleared
- S114 – CLR:PRTST, IPORT, UNPROT, CLEAR – EDGE Port, Port Failure has cleared
- S210 – CLR:TRAN, IPORT, UNPROT, CLEAR – UNI Port, IN Signal Fault has cleared
- S213 – CLR:PRTST, IPORT, UNPROT, CLEAR – UNI Port, Port Failure has cleared
- S220 – CLR:TRAN, IPORT, UNPROT, CLEAR – eNNI Port, IN Signal Fault has cleared
- S221 – CLR:PRTST, IPORT, UNPROT, CLEAR – eNNI Port, Port Failure has cleared
- S102 – CLR:TRAN, CONN, AUTONR, CLEAR – Original Path is Clear, Connection OK

- S107 – CLR:TRAN, IPORT, AUTONR, CLEAR – EDGE Port, IN Signal Fault has cleared
- S115 – CLR:PRTST, IPORT, AUTONR, CLEAR – EDGE Port, Port Failure has cleared
- S211 – CLR:TRAN, IPORT, AUTONR, CLEAR – UNI Port, IN Signal Fault has cleared
- S214 – CLR:PRTST, IPORT, AUTONR, CLEAR – UNI Port, Port Failure has cleared
- S226 – CLR:TRAN, IPORT, AUTONR, CLEAR – eNNI Port, IN Signal Fault has cleared
- S227 – CLR:PRTST, IPORT, AUTONR, CLEAR – eNNI Port, Port Failure has cleared
- S105 – CLR:TRAN, PATH, AUTORV, WTORV – Original Path is Clear, Waiting to Revert

The following notification message is valid for autoreroute revertive connections:

- S125 – CLR:RESTR, PATH, AUTORV, CLEAR – Reroute path of AUTORV is now OK
- S108 – CLR:TRAN, IPORT, AUTORV, CLEAR – EDGE Port, IN Signal Fault has cleared
- S116 – CLR:PRTST, IPORT, AUTORV, CLEAR – EDGE Port, Port Failure has cleared
- S216 – CLR:TRAN, IPORT, AUTORV, CLEAR – UNI Port, IN Signal Fault has cleared
- S217 – CLR:PRTST, IPORT, AUTORV, CLEAR – UNI Port, Port Failure has cleared
- S222 – CLR:TRAN, IPORT, AUTORV, CLEAR – eNNI Port, IN Signal Fault has cleared
- S223 – CLR:PRTST, IPORT, AUTORV, CLEAR – eNNI Port, Port Failure has cleared
- S103 – CLR:TRAN, CONN, N1PLUS1, CLEAR – N1PLUS1 Connection is OK
- S104 – CLR:TRAN, PATH, N1PLUS1, CLEAR – N1PLUS1 Path is now OK
- S109 – CLR:TRAN, IPORT, N1PLUS1, CLEAR – EDGE Port, IN Signal Fault has cleared
- S117 – CLR:PRTST, IPORT, N1PLUS1, CLEAR – EDGE Port, Port Failure has cleared
- S212 – CLR:TRAN, IPORT, N1PLUS1, CLEAR – UNI Port, IN Signal Fault has cleared
- S215 – CLR:PRTST, IPORT, N1PLUS1, CLEAR – UNI Port, Port Failure has cleared

- S224 – CLR:TRAN, IPORT, N1PLUS1, CLEAR – eNNI Port, IN Signal Fault has cleared
- S225 – CLR:PRTST, IPORT, N1PLUS1, CLEAR – eNNI Port, Port Failure has cleared
- S137 – CLR:TRAN, CONN, P1PLUS1, CLEAR – P1PLUS1 Connection is OK
- S138 – CLR:TRAN, PATH, P1PLUS1, CLEAR – P1PLUS1 Path is now OK
- S139 – CLR:TRAN, IPORT, P1PLUS1, CLEAR – EDGE Port, IN Signal Fault has cleared
- S140 – CLR:PRTST, IPORT, P1PLUS1, CLEAR – EDGE Port, Port Failure has cleared
- S135 – CLR:TRAN, CONN, YCONN, CLEAR – YCONN Conection is OK
- S136 – CLR:TRAN, PATH, YCONN, CLEAR – YCONN Path is now OK
- S118 – CLR:PRTST, IPORT, YCONN, CLEAR – EDGE Port, Port Failure has cleared
- S119 – CLR:TRAN, IPORT, YCONN, CLEAR – EDGE Port, IN Signal Fault has cleared
- S120 – CLR:TRAN, CONN, MBYNWRK, CLEAR – Original Path is Clear, Connection OK
- S121 – CLR:TRAN, IPORT, MBYNWRK, CLEAR – EDGE port, IN Signal Fault has cleared
- S122 – CLR:PRTST, IPORT, MBYNWRK, CLEAR – EDGE port, Port Failure has cleared
- S123 – CLR:TRAN, PATH, MBYNWRK, WTORV – Original Path is Clear, Waiting to Revert
- S126 – CLR:PROT, PATH, MBYNWRK, CLEAR – Protection path of MBYNWRK is now OK
- S124 – CLR:TRAN, CONN, MBYNPRT, CLEAR – MbyN Protecting Connection is OK

Path Setup Failure Notification in case of Retries

A special case of the REPT PTHEVT message covers the situation when a path setup attempt failed and a path setup retry will occur. The error message included is the code that would have been returned in a failed ENT-PATH response. The following is the output response:

```
   WS-LT-ONNS-A 2002-07-01 08:11:05
A  01104  REPT PTHEVT
   ″TIME=07-59-59,DATE=2002-08-
      01:135.13.196.45:SNN:S311::PEV:TRAN,PATH,AUTONR,FAILED:\″Path operation
      attempt failed, An error occurred\″:PathID=135.13.196.45-9874″
   ″ConnID=135.13.196.45&1-1-4-2-1&1,ConnStatus=CON&FAILED,ONNS,
   ConnAlias=AEXP-NY-BOS-ckt,
   PathID=135.13.196.45-654,State=CON&FAILED[:
```

```
   PathID=135.13.196.45-9874,State=CON&FAILED]″
   ″Error=E022:\″Status, Path Setup Failure caused by signal transmission failure at
       egress node\″:NodeID=nodeIPaddr″
;
```

Group Setup Failure Notification

A special case of the REPT PTHEVT message covers the situation when a group setup attempt is incomplete as a result of a failed transmission check (wait for light) of a worker connection of a M:N group. The error message included is the code that would have been returned in a failed ENT-PATH response. The following is the output response:

```
   WS-LT-ONNS-A 2002-07-01 08:11:05
A  01104  REPT PTHEVT
   ″TIME=07-59-59,DATE=2002-08-
      01:135.13.196.45:SNN:S311::PEV:TRAN,GRP,MBYNWRK,FAILED:\″Group operation
      attempt incomplete, An error occurred\″:GrpID=135.13.196.45-97″
   ″ConnID=135.13.196.45&1-1-4-2-1&1,ConnStatus=CON&FAILED,ONNS,
   ConnAlias=AEXP-NY-BOS-ckt,
   PathID=135.13.196.45-654,State=CON&FAILED″
   ″Error=E022:\″Status, Path Setup Failure caused by signal transmission failure at
       egress node\″:NodeID=nodeIPaddr″
;
```

- S301 – PEV:TRAN, PATH, UNPROT, DLT-FL – Path delete failed during setup retry or restoration
- S310 – PEV:TRAN, PATH,UNPROT, FAILED – ″Path operation attempt failed. An error occurred″, PathID=pathID ″Error=*Exxx*:″*errorCode,errorText*″,*errorData*″
- S302 – PEV:TRAN, PATH, AUTONR, DLT-FL – Path delete failed during setup retry or restoration
- S311 – PEV:TRAN, PATH,AUTONR, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode,errorText*″,*errorData*″
- S303 – PEV:TRAN, PATH, AUTORV, DLT-FL – Path delete failed during setup retry or restoration
- S312 – PEV:TRAN, PATH,AUTORV, FAILED – ″Path operation attempt failed. An error occurred″, PathID=pathID ″Error=*Exxx*:″*errorCode,errorText*″,*errorData*″
- S304 – PEV:TRAN, PATH, N1PLUS1, DLT-FL – Path delete failed during setup retry or restoration
- S313 – PEV:TRAN, PATH,N1PLUS1, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode,errorText*″,*errorData*″
- S145 – PEV:TRAN, PATH, P1PLUS1, DLT-FL – Path delete failed during setup retry or restoration

- S146 – PEV:TRAN, PATH,P1PLUS1, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode*,*errorText*″,*errorData*″
- S305 – PEV:TRAN, PATH, YCONN, DLT-FL – Path delete failed during setup retry or restoration
- S314 – PEV:TRAN, PATH,YCONN, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode*,*errorText*″,*errorData*″
- S306 – PEV:TRAN, PATH, MBYNWRK, DLT-FL – Path delete failed during setup retry or restoration
- S315 – PEV:TRAN, PATH,MBYNWRK, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode*,*errorText*″,*errorData*″
- S317 – PEV:TRAN,GRP,MBYNWRK,FAILED – ″Group operation attempt incomplete. An error occured.″,GrpID=connGrpID ″Error=*Exxx:*″*errorCode*, *errorText*″,*errorData*″
- S307 – PEV:TRAN, PATH, MBYNPRT, DLT-FL – Path delete failed during setup retry or restoration
- S316 – PEV:TRAN, PATH,MBYNPRT, FAILED – ″Path operation attempt failed. An error occurred.″, PathID=pathID ″Error=*Exxx:*″*errorCode*,*errorText*″,*errorData*″

The following table gives an overview of all clear event codes defined:

	UNPROT	AUTONR	AUTORV	N1PLUS1	P1PLUS1	YCONN	MBYNWRK	MBYNPRT
CLR: TRAN; CONN; <protType>, CLEAR	S101	S102	S102	S103	S137	S135	S120	S124
CLR: TRAN; PATH; <protType>, CLEAR	n/a	n/a	n/a	S104	S138	S136	n/a	n/a
CLR: TRAN; IPORT; <protType>, CLEAR	S106	S107	S108	S109	S139	S119	S121	n/a
CLR: PRTST, IPORT, <protType>, CLEAR	S114	S115	S116	S117	S140	S118	S122	n/a
CLR: TRAN, PATH, <protType>, WTORV	n/a	n/a	S105	n/a	n/a	n/a	S123	n/a

	UNPROT	AUTONR	AUTORV	N1PLUS1	P1PLUS1	YCONN	MBYNWRK	MBYNPRT
PFL: TRAN, PATH, <protType>, DLT-FL	S301	S302	S303	S304	S145	S305	S306	S307
PEV: TRAN, PATH, <protType>, FAILED	S310	S311	S312	S313	S146	S314	S315	S316
PEV: TRAN, GRP, <protType>, FAILED	n/a	n/a	n/a	n/a	n/a	n/a	S317	n/a
CLR: RESTR, PATH, <protType>, CLEAR	n/a	n/a	S125	n/a	n/a	n/a	S126	n/a

Issues wrt. event codes, that should be solved in a later release (R7):

- Currently S102 is defined for AUTONR, but should be defined for AUTORV Implementation: Simply “AUTO” is used
- The following codes are defined in SW, but not used, and also not defined in Req.
  - msgTable[NNTypes::S110] = ″S110:CLR:TRAN,EPORT,UNPROT,CLEAR,\\\\\″ EDGE port, Dest IN Signal Fault has cleared \\\\\″″″;
  - msgTable[NNTypes::S111] = ″S111:CLR:TRAN,EPORT,AUTONR,CLEAR,\\\\\″ EDGE port, Dest IN Signal Fault has cleared \\\\\″″″;
  - msgTable[NNTypes::S112] = ″S112:CLR:TRAN,EPORT,AUTORV,CLEAR,\\\\\″ EDGE port, Dest IN Signal Fault has cleared \\\\\″″″;
  - msgTable[NNTypes::S113] = ″S113:CLR:TRAN,EPORT,N1PLUS1,CLEAR, \\\\\″ EDGE port, Dest IN Sig-nal Fault has cleared \\\\\″″″;
- Not used in SW
  S110, S111, S112, S113, S114, S115, S116, S117, S118, S119

Related TL1 messages

- “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698)
- “ENT-PATH: Create ONNS path connection” (p. 3-1701)
- “ED-PATH: Add or modify alias of ONNS connection” (p. 3-739)
- “DLT-PATH: Delete ONNS path connection” (p. 3-222)
- “RTRV-NNDR: Retrieve ONNS neighbor nodal data repository” (p. 3-2561)

- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-NNCHG notification" (p. 3-2137)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)

REPT SW: Report switch

`REPT SW` messages are generated autonomously to report equipment protection switches of a duplex equipment unit pair. These equipment pairs are referred to as 1+1 equipment protection groups. The message indicates the equipment/circuit pack that was placed in the active state.

The notification has been extended for LOXC equipment protection groups.

Output format

The general format of a REPT SW message is as follows:

```
   tid date time
A  atag REPT SW
   ″actid,stbyid″
;
```

Output parameters

Command parameters

1. **`atag`** – AutomaticTag
   This parameter specifies the message sequence number. The number is incremented by one for each autonomous message generated by the NE. Leading zeros are not suppressed.
   Type: Numeric string.
   Range: 000 .. 999
2. **`actid`** – Active Identifier
   This parameter specifies the circuit pack AID which is active within the 1+1 equipment protection group.
   Type: AID
   Range: CpAID (if reported by a EQP group with protype unequal to EQPTLOXC) or LoxcAID (if reported by a EQP group with protype equal to EQPTLOXC).
3. **`stbyid`** – Standby Identifier
   This parameter specifies the circuit pack AID which is standby (non-active) within the 1+1 equipment protection group.
   Type: AID
   Range: CpAID (if reported by a EQP group with protype unequal to EQPTLOXC) or LoxcAID (if reported by a EQP group with protype equal to EQPTLOXC).

Detailed behavior description

The system shall evaluate the current fault information and the current active unit at the protection group creation time. Based on that, the sytem shall calculate and report the selected active unit and the standby unit.

Example input/output

The following example shows a switch report (with *stbyid* parameter):

```
   LT-UNITE 99-01-10 08:12:00
A  001 REPT SW
   ″1-1-#-#-xcp-cp,1-1-#-#-xcw-cp″
;
```

Related TL1 messages

- “RTRV-PROTN-GRP: Retrieve protection group” (p. 3-2759)
- “OPR-PROTNSW: Operate protection switch” (p. 3-2057)
- “RLS-PROTNSW: Release protection switch” (p. 3-2203)

References

GR-833-CORE

RLS-EXT-CONT: Release external control

The **RLS-EXT-CONT** command releases an MDO port (Miscellaneous Discrete Output / external discrete control), i.e. switches off the MDO port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **RLS-EXT-CONT**:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDO port entities on which the command operates. If misc_outall is specified the configuration of all MDO's will be retrieved.
   Type: AID
   Range: Miscellaneous_out_discreteAID
   Default: misc_outall (if the *aid* is omitted).
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

The following example shows the successful completion of the **RLS-EXT-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

If the **RLS-EXT-CONT** command does not alter the existing attributes, i.e., the control referenced by the aid is already released, the NE will not deny the command. The system will respond with the completion message shown above.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

If no probable cause is assigned to the MDO port the command changes the state of the MDO specified via aid to ″switched off″, i.e. the MDO port will be deactivated.

The update will be reported via a RLS-EXT-CONT – REPT DBCHG message. Refer to the definition of the RLS-EXT-CONT notification.

Example input/output

The following example shows the successful completion of the **RLS-EXT-CONT** command by the NE:

```
RLS-EXT-CONT:LT-UNITE-123:MISC_OUT3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RLS-EXT-CONT** command.

Validation rule: probable cause assigned to MDO port

A probable cause is assigned to the MDO port. This command will be not performed.

Error code: `SDNC – Status, Data Not Consistent, asgncond not empty`

Related TL1 messages

- "RTRV-ATTR-CONT: Retrieve attribute control" (p. 3-2254)
- "SET-ATTR-CONT: Set attribute control" (p. 3-3133)
- "RLS-EXT-CONT: Release external control" (p. 3-2190)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

RLS-EXT-CONT notification

Output format

The **`RLS-EXT-CONT`** notification has the following specific syntax:

`RLS-EXT-CONT`:*aid*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This is the address of the MDO port entity for which the configuration is retrieved or on which an operation is performed.
   Type: AID.
   Range: Miscellaneous_out_discreteAID. misc_outall must not be reported or used.

Example output

The example shows a normal RLS-EXT-CONT notification:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:RLS-EXT-CONT:MISC_OUT3”
;
```

RLS-LPBK (cross-connection loopback) notification

There will be no **RLS-LPBK** notification because the changes (for the parameter *crslpbkstat* and *crslpbkrate*) caused by an RLS-LPBK command will be notified via an ED-*rr* (*rr* being either VC3 or STS1) notification for normal tributaries and by an ED-VCGTRIB notification for virtual concatenation tributaries.

RLS-LPBK (facility loopback) notification

There will be no **`RLS-LPBK`** notification because the changes (for the parameter *lpbk* and *fsflpbkstat*) caused by an RLS-LPBK command will be notified via an ED-rr notification for an OP***n*** / EP***n*** / OPT***n*** port (rr specifying the affected port i.e. OC<M>/STM<N>OC<M>T/STM<N>T) or via an ED-EPORT notification for a GE***n*** port (*fsflpbkstat* only).

RLS-LPBK: Release loopback (cross-connection)

The **RLS-LPBK**-*modifier* command can be initiated by a user to release a loopback on a tributary.

In this section only the loopback on tributaries are considered.

The *modifier* in the **RLS-LPBK** command indicates the rate on which the loopback command is applied to. There is no difference if the rate is specified as a SONET rate (STS1, STS3,..) or as SDH-rate (VC3, VC4,..) for the i.e. the RLS-LPBK-<*sonet-rate*> yields to the same physical and logical behavior as RLS-LPBK-<*sdh-rate*>. No check is made if the *modifier* matches the interface-standard of the port i.e. if the port is provisioned as SONET (resp. SDH) and the rate is a SONET (resp. SDH) rate.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **RLS-LPBK**-*modifier*:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the tributary- respectively cross-connection-rate on which the **OPR-LPBK** command acts on.
   For detailed description please refer to the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48

- STS192
- LOVC3
- LOVC12
- LOVT1.

No use of the **all** value is allowed.

2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   This parameter specifies the tributary for which the loopback is released.
   Type: GenTribAID or GenLOTribAID
   Range: No **all** or **usall** or **vall**-value is possible for any part of the aid.
4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Output format

If the **RLS-LPBK** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

Existing loopback on tributaries can be released via the **RLS-LPBK** command. The attribute *crslpbkstat* of the entity tributary is updated to NO.

This update is reported by

- an ED-*rr* (*rr* either VC3 or STS1) notification for normal tributaries
- an ED-VCGTRIB notification for virtual concatenation tributaries
- an ED-PTF notification for HO path terminations
- an ED-LONIM notification for low order tributaries.

Updated affected entities when releasing a HO-Loopback

Apart from the tributary entity no further entities are affected by the **RLS-LPBK** command.

Updated affected entities when releasing a LO-Loopback

In case of releasing a LO-loopback which leads to a removal of the last LO-Cross-connection within the related HO-tributary the substructuring of this HO-tributary may be removed automatically.

Example input/output

```
RLS-LPBK-vc3:UNITE-NE:1-1-#-#-1-1-4:123456;

   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Valid rate

Loopback command processing verifies that the specified rate is equivalent to the rate of the existing loopback. Hereby the SONET-rates are treated identical to the SDH-rates (i.e. sts1/vc3, STS3/vc4 etc.).

Error code:

`IDNC – Input Data Not Consistent, Rate is inconsistent with existing cross-connection`

Validation rule: Entities (tributaries) must exist

Loopback command processing verifies that the logical tributaries specified in the command do exist.

Error code:

`IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist`

Validation rule: Loopback must exist

Loopback command processing verifies that the **`RLS-LPBK`** command can not be applied to non-existing loopback.

Error code: `SNVS – Status, Not in Valid State, loopback does not exist`

Related TL1 messages

- “OPR-LPBK: Operate facility loopback” (p. 3-2044)
- “RTRV-LPBK (facility): Retrieve loopback (facility)” (p. 3-2511)

References

GR-199-CORE

RLS-LPBK: Release loopback (facility)

The **RLS-LPBK**-*modifier* command can be initiated by a user to release an existing loopback on an OP***n***, GE***n***, EP***n*** or OPT***n*** port in the NE.

Note that the loopback type is not specified explicitly as a command parameter – instead the currently existing loopback (either a near-side or a far-side) will be released.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T4

Priority: 1

Abortable: No

Input format

Command syntax: **RLS-LPBK**-*modifier*:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **modifier**
   The modifier parameter indicates the port rate on which the **OPR-LPBK** command acts on.
   Type: Alphanumeric value set.
   Range:
   - STM1E
   - STM1
   - STM4
   - STM16
   - STM16T
   - STM64
   - STM256
   - OC3
   - OC12
   - OC48
   - OC48T
   - OC192
   - OC768
   - 1GE

- 1GEPL
- 10GEPL
- T3.

No use of the **`all`** value is allowed.

2. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **`aid`** – Access Identifier
   Access Identifier of the port for which the loopback state is returned.
   Type: AID
   Range: SPortAID (when using OP***n***, EP155, OPT***n*** packs or EC1 interfaces on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_SPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   APortAID (when using DS3 interfcaes on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_APortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   EPortAID (when using GE***n*** packs) as defined in AID structure. The values:
   - all for Simple_EPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.
4. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Output format

If the **`RLS-LPBK`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

Behavior – OP*n* Port

Existing loopback on ports can be released via the **RLS-LPBK** command. The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by a ED-*rr* notification for an OP***n*** port (*rr* specifying the affected port i.e. OC<M>/STM<N>).

Example:

RLS-LPBK-OC48:UNITE-NE:1-1-#-#-1-1:123456;

REPT DBCHG

ED-OC48:1-1-#-#-1-1::fsflpbkstat=NO;

Behavior – OPT*n* Port

Existing loopback on ports can be relaesed via the **RLS-LPBK** command. The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by a ED-*rr* notification for an OPT***n*** port (*rr* specifying the affected port i.e. OC<M>T/STM<N>T).

Behavior - EP155 Port

Existing loopback on ports can be released via the “RLS-LPBK: Release loopback (facility)” (p. 3-2198) command.

The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by an ED-rr notification for an EP155 port (*rr* specifying the affected port STM1E).

Behavior – Asynchronous port

Existing loopback on asynchronous ports can be released via the **RLS-LPBK** command. The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by an **ED-*rr*** notification for an asynchronous port (*rr* specifying the affected port). Not all types of asynchronous ports support loopbacks. See the modifier range for supported asynchronous ports.

Behavior - EC1 Port

Existing loopback on EC1 ports can be released via the **`RLS-LPBK`** command. The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by an ED-EC1 notification for the port.

Behavior – GE1 Port

Existing loopback on ports can be released via the **`RLS-LPBK`** command. The attribute *lpbk* and *fsflpbkstat* of the entity port is updated by this command:

- *lpbk* should be set to NO in case the *lpbktyp* = FACILITY is released
- *fsflpbkstat* should be set to NO in case the *lpbktyp* = FSF is released.

This update is reported by an ED-EPORT notification for a GE***n*** port (only *fsflpbkstat*).

Behavior – Any Port on GE10PL1

Loopback on all successfully provisioned ports of a GE10PL1 pack can be switched via the **`OPR-LPBK`** command.

The parameter fsflpbkstat of the entity EPORT is updated by this command:

- *fsflpbkstat* should be set to YES in case the *lpbktyp* = FSF.

This update is reported by an ED-EPORT notification for the related EPORT (only for the parameter *fsflpbkstat*). Any attempts to change the parameter *lpbktyp* to the value FACILITY is rejected.

Updated affected entities

Apart from the port entity no further entities are affected by the **`RLS-LPBK`** command. Updates of the port entities are reported via ED-rr (OP***n*** port/ EP***n*** port / OPT***n*** port) or ED-EPORT (GE***n*** port) notifications.

Example input/output

Please refer to the *Detailed behavior description* section for this command.

Error responses

Validation rule: Loopback must exist

Loopback command processing verifies that the **`RLS-LPBK`** command can not be applied to non-existing loopback. This means e.g. that a request to release a near-side (far-side) loopback while no near-side (far-side) loopback exists is rejected.

Error code: `SNVS – Status, Not in Valid State, loopback does not exist`

Validation rule: *modifier* consistent with given port-rate

Loopback command processing verifies that the *modifier* parameter is consistent with the rate of the port (given by the *aid* parameter). E.g. for an OP2G5 only the *modifier* values STM16 or OC48 (depending on the port's interface standard) are allowed, for an OP10 only *modifier* values STM64 or OC192 (depending on the port's interface standard) are allowed etc.

Error code:

`IDNC – Input Data Not Consistent, modifier is inconsistent with existing port rate`

Validation rule: *modifier* consistent with given port's interface standard

Loopback command processing verifies that the *modifier* parameter is consistent with the interface standard of the port (given by the *aid* parameter). E.g. for an SDH port only the *modifier* values for STM16 or STM64 or STM256 are valid, for a SONET port only the *modifier* values OC48 or OC192 or OC768 are valid.

Error code:

`ICNV – Input Command Not Valid, Interface Standard does not match`

Validation rule: *modifier* consistent with the related circuit pack type

Loopback command processing shall verify in case that the modifier parameter does not distinguish various entities by transmission rate or by interface standard that the addressed entity belongs to the correct pack type. E.g. if the modifier value 1GE has been sent for an 1 GE port on the GE10PL1 pack or if the modifier value 1GEPL has been sent for a port on the GE1 pack, then the command is rejected.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with the related circuit pack type`

Related TL1 messages

- "OPR-LPBK: Operate facility loopback" (p. 3-2044)
- "RTRV-LPBK (facility): Retrieve loopback (facility)" (p. 3-2511)

References

GR-199-CORE

RLS-PROTNSW: Release protection switch

The **`RLS-PROTNSW`** command allows to clear protection switch requests for one or more existing protection group instances of any protection group type. Different identification methods are supported for the different protection types.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No

Input format

1+1 equipment group

Command syntax: **`RLS-PROTNSW`**:*tid*:*aid*:*ctag*::*protype*:*spec_block*;

Multiplex section/line protection group

Command syntax: **`RLS-PROTNSW`**:*tid*:*aid*:*ctag*::*protype*:*spec_block;*

2-/4-fibre MS-SPRing/BLSR protection group

Command syntax: **`RLS-PROTNSW`**:*tid*:*[aid]*:*ctag*::*protype*,*[rid]*:*spec_block*;

Path protection group

Command syntax: **`RLS-PROTNSW`**:*tid*:*[aid]*:*ctag*::*protype, ,[ppgname][,aid1]*:*spec_block*;

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.

> **Important!** The use of *ppgname* as input parameter with alias function is not yet supported. Address path-protected cross-connections individually via *Navis*® Optical EMS/CIT GUI using their AIDs.

> **Important!** If a RLS-PROTNSW command is executed on an MSP 1+1 optimized protection group while the command would have no effect (e.g. already cleared by previous command), then the command is denied with the error code: `SROF, Status, Requested Operation Failed`. The correct behavior should be that the command completes without error.

Input parameters

Command parameters (1+1 equipment protection group)

1. **tid** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Description: The parameter specifies the aid of 1+1 equipment protection group.
   Type: AID
   Range: SwitchAndTmgProtGrpAID or CtlProtGroupAID or EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID
   All actual existing equipment protection groups will be returned with a specific AID:
   - SwitchAndTmgProtGrpAID for high order switch fabric and timing group,
   - CtlProtGroupAID for system controller group,
   - EP155ProtGroupAID for electrical 155 MBit/s groups,
   - EP51ProtGroupAID for electrical 51 MBit/s groups,
   - LOXCProtGroupAID for low order switch fabric groups.
   - TMUXProtGroupAID for TransMUX groups
   - ES64UProtGroupAID for es64u groups

   Default: No default
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **protype** – Protection Type
   The parameter specifies the type of the protection group.
   Type: Alphanumeric value set.
   Range:
   - EQPTSWFBR (if *aid* is SwitchAndTmgProtGrpAID)
   - EQPTCTL (if *aid* is CtlProtGroupAID)
   - EQPTEP155 (if *aid* is EP155ProtGroupAID)
   - EQPTEP51 (if *aid* is EP51ProtGroupAID)
   - EQPTLOXC (if *aid* is LOXCProtGroupAID)
   - EQPTTMUX (if *aid* is TMUXProtGroupAID)
   - EQPTES64U (if *aid* is ES64UProtGroupAID)

Command parameters (multiplex section/line protection group)

1. **`tid`** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port aid being a member of the group.
   Type: AID.
   Range: OnePlusOne_ProtGrpAID (not ranged) or SPortAID NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged).
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
4. **`protype`** – Protection Type
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later).
   Type: Value set.
   Range: 1+1 or 1xNOPT

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **`tid`** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F
5. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: If *aid* is specified, then *rid* can be omitted.

Command parameters (Path protection group)

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies a path protection group *aid* or an input tributary *aid* being a member of the group in combination with *ppgname*.
   The parameter specifies an input tributary when used in combination with parameter *aid1*.
   Option: If *ppgname* is specified, then *aid* can be omitted or used as an input tributary *aid*. The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended. For low order path protections, the *aid* is mandatory since the *ppgname* is not supported for those protection groups. For such protection groups, always a path protection group AID has to be specified.
   Option: If parameter *aid* specifies a HO tributary on a port of port class EDGE and this tributary is combined with another HO tributary within an OUC, then the command shall operate on all protection groups of type PATHDRI identified by these two HO tributaries, if existing. If *aid* specifies a HO tributary on a port of port class INNI, the command is rejected.
   Option: When parameter aid is used in combination with parameter *aid1*, then the command shall operate on all protection groups of type PATHDRI having the AIDs given in parameters *aid* and *aid1* as input tributaries. If *aid* or *aid1* specifies a tributary on a port of port class INNI, the command is rejected.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.

3. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: PATHDRI
4. **`ppgname`** – Path protection group Name
   In particular follow the reference to check options and mandatory omissions associated with this parameter.
   The parameter specifies the path protection group name. Storage and retrieval of the *ppgname* is case sensitive, but actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
   Option for high order path protection groups: For high order path protections, if *aid* is specified, then *ppgname* can be omitted. For low order path protections, the *ppgname* is omitted as not being supported at all.
   If an input value is specified for *ppgname* and none is specified for aid, then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified path protection group name. If input values are specified for aid and *ppgname* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified *aid*, *ppgname* pair.
   The command will complete successfully if there is no protection group described by *aid*, *ppgname*, but no data will be returned.
   The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Type: text string
   Range: 1 to 24 characters
   Option: or high order path protections, if *aid* is specified, then *ppgname* can be omitted. For low order path protections, the *aid* is not omitted since the *ppgname* is not supported for those protection groups. For lower order path protection groups, always a path protection group AID has to be specified. If an input value is specified for *ppgname* and none is specified for *aid*, then the command operates on all protection groups that have the specified path protection group name. If input values are specified for aid and *ppgname* then the command operates on all protection groups that have the specified *aid*, *ppgname* pair. The command will complete successfully if there is no protection group described by *aid*, *ppgname*.
5. **`aid1`** – Access identifier 1
   Together with parameter *aid* this parameter specifies a pair of tributaries in order to execute switches to emulated selector bridges in one command.
   Option: If input values are specified for *aid* and *aid1* then OPR-PROTNSW operates on all protection groups of type PATHDRI that have these specified tributaries as inputs. If input values are specified for *aid* and *aid1*and additionally a

value is specified for *ppgname*, OPR-PROTNSW operates only on those protection groups that additionally match the ppgname value. If aid1 specifies a HO tributary in a port of port class INNI, the command is rejected.
Type: AID
Range: STS1NumAID

Command parameters (Constituent path protection group)

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   The parameter specifies the constituent member of a path protection group or an input tributary *aid* of a constituent member within the path protection group in combination with *ppgname*.
   If *ppgname* is specified, then aid can be used as an input tributary aid. The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Type: AID
   Range: STS1NumAID, VCGNumAID
   The use of wild cards is not restricted for STS1NumAID and VCGNumAID. No wildcards are allowed in a GenLOTribAID.
   This range specifies all AIDs, which can be used to identify a constituent path protection group.
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: CONSTITUENTPATH

Spec block parameters (1+1 equipment, path protection group)

1. **`sc`** – Switch Command
   This parameter specifies the clear command to be executed by the protection state machine.
   Type: Alphanumeric value set.
   Range: CLEAR
   Default: CLEAR
   Option: If the parameter is omitted the default switch command will be executed.

Spec block parameters (multiplex section/line protection group)

1. **sc** – Switch Command
   This parameter specifies the clear command to be executed by the protection state machine.
   Type: Alphanumeric value set.
   Range: CLEAR
   Default: CLEAR
   Option: If the parameter is omitted the default switch command will be executed.

Spec block parameters (2/4-fibre MS-SPRing/BLSR protection group)

1. **destside** – Destination Side

   This parameter specifies the destination side to which the switch command is applied; i.e.

   - it is EAST for the east ring side
   - it is WEST for the west ring side

   Type: Alphanumeric value set.
   Range: EAST, WEST
2. **sc** – Switch Command
   This parameter specifies the clear command to be executed by the protection state machine.
   Type: Alphanumeric value set.
   Range: CLEAR
   Option: If the parameter is omitted the default switch command will be executed.

Output format

After receiving the **RLS-PROTNSW** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Detailed behavior description

The command clears the protection switch of the identified protection groups. The clear command is processed according to the state machine behavior for the different protection group types. As consequence, the values of the following attributes of the state machine may be updated:

EQPTSWFBR: *actunit, swreq*

EQPTCTL: *swreq*

EQPTEP155: *actunit, swreq*

EQPTEP51: *actunit, swreq*

EQPTLOXC: *actunit*, *swreq*

EQPTTMUX: *actunit*, *swreq*

EQPTES64U: *actunit*, *swreq*

1+1/1:1: *actunit*, *swreq*

2F: *eswreq*, *wswreq*

4F: *eactunit*, *eswreq*, *wactunit*, *wswreq*,

PATHDRI: *actunit*, *actunitaid*, *swreq*

The update of the attributes is reported by means of a protection switch notification for the corresponding protection group type.

Behavior of path-protected pipe cross-connections

If the protection type specified in the command is PATHDRI, then the command is applied to all constituent members of the path-protected pipe cross-connection. The system shall send protection switch notifications only for those constituent members, where the command leads to a change of the protection state.

Boundary Independence for Constituent Members:

If the protection type specified in the command is CONSTITUENTPATH, then the command is applied to the constituent member of the path-protected pipe cross-connection, which includes the specified tributary AID.

Identification of path protection groups

A set of path protection groups is identified by a path protection group name (*ppgname*). If an input tributary AID (*aid*) is specified in addition to the *ppgname*, then the command operates on the subset of protection groups that share both of these parameters.

The protection group name handling shall only apply to protections groups which have tributaries of high-order signal rates been involved.

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.

For PATHDRI, the protection switch applies to all constituents of the identified tributary. For this case, the following table summarizes the use of path protection group names as related to RLS-PROTNSW.

aid	ppgname	comment
not used	ppgname	operation applies to all tributaries that are group members of the ppgname (note 1)

aid	ppgname	comment
grp member (input) logical tributary AID	ppgname	operation applies to all tributaries that are group members of the aid, ppgname subset
path prot grp id (output logical tributary AID)	not used	normal, non-aliasing target identification

Note 1: group members are defined to be the working and protection input logical tributaries of the path protection group.

Example input/output

1+1 Equipment Protection Group

The following example shows an successful clear command for a 1+1 equipment protection group:

```
RLS-PROTNSW:LT-UNITE:1-1-estgrp:123456::EQPTSWFBR;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Multiplex section/line protection group (1+1)

The following example shows an successful clear command for a 1+1 multiplex section/line protection group:

```
RLS-PROTNSW:LT-UNITE:1-1-o110:123456::1+1;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

2 fibre MS-SPRing/BLSR protection group

The following example shows an successful ’clear’ command for the east side of a 2F MS-SPRing/BLSR protection group:

```
RLS-PROTNSW:LT-UNITE:1-1-t001:123456::2F:sc=CLEAR,destside=EAST;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

4 fibre MS-SPRing/BLSR protection group

The following example shows an successful ’clear’ command for the west side of a 4F MS-SPRing/BLSR protection group:

```
RLS-PROTNSW:LT-UNITE:1-1-f001:123456::4F:sc=CLEAR,destside=WEST;
```

```
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Path protection group

The following example shows an successful clear command for a path protection group:

```
RLS-PROTNSW:LT-UNITE:1-1-#-#-10-1-25:123456::PATHDRI;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

The following example shows a successful clear command to both path protection groups forming an emulated selector bridge on input tributaries 1-1-#-#-8-1-25 and 1-1-#-#-6-1-25:

```
RLS-PROTNSW:LT-UNITE:1-1-#-#-8-1-25:123456::PATHDRI,,,
1-1-#-#-6-1-25:;

   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses (general)

Validation rule: consistent *aid* and *protype*

The specified *aid* is consistent with the specified *protype*.

Error code: `IIAC – Input, Invalid ACcess identifier,incorrect syntax`

Error responses (multiplex section/line protection group)

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Error responses (2-/4-fibre MS-SPRing/BLSR protection group)

Validation rule: consistent aid and rid

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IIAC – Input, Invalid ACcess identifier,incorrect syntax`

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Error responses (Path protection group)

Validation rule: Protection Group must exist

If the parameter *aid*, or the combination of the parameters *aid* and *ppgname* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Protection Group must exist

If the combination of the parameters *aid* and *aid1* does not specify a path protection group, then the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

Validation rule: succeeding switch command

The switch command is executed by the protection state machine running on the XC pack.

Error code: `SPFA – Status, Protection unit FAiled`

Validation rule: *ppgname* and low order path protections

If a ppgname is given as input parameter while the aid references a low order tributary, then the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, ppgname not supported for this transmission rate`

Validation rule: Access via tributary on INNI port

If in input parameter *aid* or *aid1* for access via input or output a tributary is given located on a port of *portclass* INNI, then the following error response is returned:

Error code:
`IDNV – Input, Data Not Valid, No Access via AID located on INNI port`

Related TL1 messages

- “RTRV-PROTN-GRP: Retrieve protection group” (p. 3-2759)
- “ED-PROTN-GRP: Edit protection group” (p. 3-803)
- “ENT-PROTN-GRP: Enter protection group” (p. 3-1745)

- "ENT-CRS: Enter cross-connection" (p. 3-1464)
- "OPR-PROTNSW: Operate protection switch" (p. 3-2057)

RLS-SYNCNSW: Release synchronization switch

The **RLS-SYNCNSW** command can be initiated by user to request the management system to release (clear) any existing lockout, forced or manual timing reference switching or external output timing switch or to release the forced request on the timing mode so that it will return to the provisioned locked mode from the holdover mode.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M4

Priority: 1

Abortable: No.

Input format

Command syntax: **RLS-SYNCNSW**-*modifier*:*tid*::*ctag*::*[dest]*;

Input parameters

Command parameters

1. **modifier** – modifier
   This parameter indicates the management system on which the switch request acts. The *modifier* REF specifies the assigned timing reference switch. The *modifier* MOD specifies the clock mode switch of the system clock. The *modifier* OUTREF specifies the external timing output switch (for SDH only).
   Type: Alphanumeric value set.
   Range:
   - REF
   - MOD
   - OUTREF

   Default: Not applicable.
2. **tid** – target identifier
   See definition of *tid – target identifier* in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – correlation tag
   See definition of *ctag – correlation tag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

4. **`dest`** – destination
   When the parameter dest is omitted with modifier REF or OUTREF the system clears a manual or forced timing reference or external output timing switch request state.
   This parameter specifies the assigned timing reference identifier destination. This is the list of timing references available to the system clock or to the timing link switch of the external output timing. This parameter is valid and required for the REF and OUTREF modifiers.
   Type: AID
   Range:
   - *modifier* REF: External_timing_ref_AID, Line_timing_ref_AID
   - *modifier* OUTREF (for SDH only): Line_timing_ref_AID

   Default: Not applicable.

Output format

After receiving the **`RLS-SYNCNSW`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The **`RLS-SYNCNSW`** command generates no **`REPT DBCHG`** Report Database Change notification. The **`RLS-SYNCNSW`** command returns a REPT EVT Report Event notification in consequence of a protection switch notification.

Clear clock backup request

The system clock leaves the Forced Holdover state by means of the **`RLS-SYNCNSW`** command with *modifier* MOD without parameter. The Clear Mode Switch request removes any active external request and does not affect the state of the automatic protection switching based on synchronization reference failure. The following parameter is affected by release of the clock mode switch Forced Holdover: – clkmod_stat = NORMAL.

Clear system timing reference or output timing switch request

The Clear Reference Switch command removes any active external request as is clear the Lockout, Forced Switch and Manual Switch conditions. The Clear Reference Switch command does not affect the state of the automatic protection switching based on synchronization reference failure, except for its effect on Wait to Restore. The Clear

Reference Switch invokes the management system to terminate the wait to restore period for the specified timing reference and may be used to clear Wait to Restore. The signal status will revert to NORMAL.

Forced Switch: The system leaves the Forced Switch state by means of the **OPR-SYNCNSW** command with *sc* value LOCKOUT, or RLS-SYNCNSW command.

Manual Switch: The system leaves the Manual Switch state by means of the **OPR-SYNCNSW** command with *sc* value LOCKOUT, or OPR-SYNCNSW command with *sc* value FRCD, or RLS-SYNCNSW command.

Lockout: The system leaves the Lockout state by means of the **RLS-SYNCNSW** command. The following parameters are affected by release of the system timing reference switch:

- active_ref = [External_timing_ref_AID, Line_timing_ref_AID]
- refsw_stat = NO-REQUEST
- ref_lock = NO-LOCKOUT
- ref_stat = NORMAL

The following parameters are affected by release of the external output timing switch request:

- active_oref_stat = Line_timing_ref_AID
- orefswr_stat = NO-REQUEST
- sclkref_lockstat = NO-LOCKOUT

Example input/output

The following is an example of the **RLS-SYNCNSW** command for System Timing with *modifier* MOD:

```
RLS-SYNCNSW-MOD:LT-UNITE::123456::;

   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **RLS-SYNCNSW** command for System Timing with *modifier* REF:

```
RLS-SYNCNSW-REF:LT-UNITE::123456::EXTREF1;

   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **RLS-SYNCNSW** command for Output Timing with *modifier* OUTREF:

```
RLS-SYNCNSW-OUTREF:LT-UNITE::123456::;
```

```
   LT-UNITE 00-11-09 12:00:00
M  123456 COMPLD
;
```

Related TL1 messages

- “OPR-SYNCNSW: Operate synchronization switch ” (p. 3-2077)
- “SET-SYNCN: Set synchronization” (p. 3-3158)
- “RTRV-SYNCN: Retrieve synchronization” (p. 3-2955)

References

GR-833-CORE

RTRV-ACTL-CONT: Retrieve actual condition types

The **RTRV-ACTL-CONT** command can be initiated by users to retrieve the available condition types which can be assigned to an miscellaneous output control (MDO port / external discrete control).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ACTL-CONT**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

The following example shows the successful completion of the **RTRV-ACTL-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
   ″pftype:condtype:condinfo″
   ...
   ″pftype:condtype;condinfo″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.

Range:

- DCN: Data Communications Network
- DS1: DS1 asynchronous signals
- DS3: DS3 asynchronous ports
- ENET: Ethernet
- ENV: Environmental (Miscellaneous Discrete)
- EQUIP-PROT: Equipment with Protection State Dependence
- EQUIP-UNPROT: Equipment unprotected
- MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
- MSP-APS: Automatic protection switch
- OTN: OTN ports.
- PATH: Path MSA/NIM related
- PATHTERM: Path Termination
- PORT: SDH/SONET ports
- SCN: SCN
- SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

2. **`condtype`** – Condition Type
   This parameter identifies the type of alarm indication being reported.
   Type: Alphanumeric value set.
   Range: All defined probable causes..
3. **`condinfo`** – Information Of The Condition
   This is a textual description of the condition.
   Type: inner string of the format: \″<category>, \<description>″
   Range: category: is the category; description: is the alarm description, the string is up to 32 characters.

Detailed behavior description

The **`RTRV-ACTL-CONT`** command does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of the **`RTRV-ATCL-CONT`** command by the NE if the aid is omitted:

```
RTRV-ACTL-CONT:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
```

```
M  123456 COMPLD
   ″PORT:OCHLOF: \″Communication-Transport, OCh Loss of Frame\″″
   ″PORT:PREDCMLOS: \″Communication-Transport, Pre DCM Signal Loss\″″
         ...
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ATCL** command.

Related TL1 messages

- “RTRV-ATTR-CONT: Retrieve attribute control” (p. 3-2254)
- “SET-ATTR-CONT: Set attribute control” (p. 3-3133)
- “OPR-EXT-CONT: Operate external control” (p. 3-2032)
- “RLS-EXT-CONT: Release external control” (p. 3-2190)
- “RTRV-ALM: Retrieve alarm” (p. 3-2231).

RTRV-ACTLCP: Retrieve acceptable circuit pack list

The **RTRV-ACTLCP** command returns all circuit packs (name and qualifier) which are supported in the system; i.e. there is at least one slot, where a circuit pack of that type is accepted to be provisioned/pre-provisioned.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ACTLCP***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-ACTLCP command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

The output shall contain one line per supported circuit pack (name, qualifier pair).

Output parameters

Command parameters

1. **aid** – Access Identifier:
   See definition of *aid* input parameter in Input parameter section above.

Spec block parameters

1. **actlname** – Acceptable Circuit Pack Name
   This parameter specifies the name of a circuit pack supported by system.
   Type: value set.
   Range: 12 character.
2. **actlqual** – Acceptable Circuit Pack Qualifier
   This parameter specifies the qualifier of a circuit pack supported by system.
   Type: value set.
   Range: 7 character.

Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

- alphanumeric characters or the special character '-' only
- special characters like '/' and '.' to be omitted.

Example input/output

```
RTRV-ACTLCP:UNITE-NE:system:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″system:actlname=OP2G5,actlqual=13SR4″
   ″system:actlname=OP10,actlqual=15LR1″
    ...
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also pertain to the **RTRV-ACTLCP** command.

Related TL1 messages

- "ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
- "DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
- "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400)

RTRV-ALL: Retrieve all (port)

The **RTRV-ALL** command is used to retrieve all parameters related to SDH/SONET ports.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ALL**:*tid:aid:ctag[:::spec_block];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   AID of SDH or SONET port.
   Type: AID.
   Range: SPortAID of STMn ports and OCn ports of supported rates or EC1 port. The keyword **usall** and **all** is allowed for Simple_SlotAID. The keyword **all** is allowed for Simple_SPortAID.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not.

   The following output parameters of a port are affected:

   - strc
   - strcst.

   Type: Alphanumeric value set.
   Range: YES: Report dynamic parameters in the reply as well; NO: Do not report dynamic parameters in the reply.
   Default: NO.

Output format

In response to a valid RTRV-ALL command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of the SDH or SONET port.
   Type: AID.
   Range:
   - SPortAID for STM***n*** ports and OC***n*** ports of supported rates or EC1 port.
   - APortAID for DS3 ports.
   - TMuxAPortAID, TMuxSPortAID for TransMUX ports.

   The value **`all`** is not allowed.

Spec block parameters

1. **`ptype`** – Port Type
   This identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range:
   - STM1E
   - STM1
   - STM4
   - STM16
   - STM16T
   - STM64
   - STM256
   - OC3
   - OC12
   - OC48

- OC48T
- OC192
- OC768
- EC1
- T3
- TMAT3
- TMST3

2. <others>

Detailed behavior description

The command retrieves the parameters as described above.

The **RTRV-ALL** command does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a RTRV-ALL command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-ALL:LT-UNITE:1-1-#-#-usall-all:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-4-1:ptype=OC768,frcdus=DISABLE,sfthr=-3,sdthr=-6, pmode=MON,.....
      pmmsneses=15,pmrsneses=30″
   ″1-1-#-#-5-1: ptype=STM64,frcdus=DISABLE,dexcthr=-3,degthr=-6,
      pmode=MON,.....,pmmsneses=15,pmrsneses=30″
;
```

Error Responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ALL** command.

Related TL1 messages

- “RTRV-STM1E: Retrieve STM1E port” (p. 3-2849)
- “RTRV-STM1: Retrieve STM1 port” (p. 3-2863)
- “RTRV-STM4: Retrieve STM4 port” (p. 3-2881)
- “RTRV-STM16: Retrieve STM16 port” (p. 3-2899)
- “RTRV-STM16T: Retrieve STM16T port” (p. 3-2917)
- “RTRV-STM64: Retrieve STM64 port” (p. 3-2926)
- “RTRV-OC3: Retrieve OC3 port” (p. 3-2602)

- "RTRV-OC12: Retrieve OC12 port" (p. 3-2620)
- "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)
- "RTRV-OC48T: Retrieve OC48T port" (p. 3-2656)
- "RTRV-OC192: Retrieve OC192 port" (p. 3-2664)
- "RTRV-EC1: Retrieve EC1 port" (p. 3-2332)
- "RTRV-T3: Retrieve DS3 port" (p. 3-2968)
- "RTRV-TMAT3: Retrieve asynchronous adjunct port of the TransMUX card" (p. 3-3029)
- "RTRV-TMST3: Retrieve synchronous adjunct port of the TransMUX card" (p. 3-3034)

References

GR-199-CORE (RTRV-*rr*)

RTRV-ALL: Retrieve all (tributary)

The **RTRV-ALL** command is used to retrieve all parameters related to the tributaries.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ALL**:*tid*:*aid*:*ctag*[:::spec_block];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   AID of SDH/SONET tributary. The keyword **all** is allowed on slot, port and tributary level.
   Type: AID
   Range: STS1NumAID
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameter

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not.

   The following output parameters of a tributary are affected:

   - ptrc
   - ptrcst
   - siglb
   - trailstate.

   Type: Alphanumeric value set.
   Range: YES, report dynamic parameters in the reply as well; NO, do not report dynamic parameters in the reply.
   Default: NO.

Output format

In response to a valid RTRV-ALL command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   AID of the SDH/SONET tributary. The keyword **`all`** is not allowed on any level.
   Type: AID.
   Range: STS1NumAID.

Spec block parameters

For the spec block parameters refer to the descriptions of the following commands “RTRV-VC3: Retrieve VC3 tributary” (p. 3-3069) and “RTRV-STS1: Retrieve STS1 tributary” (p. 3-2946).

Detailed behavior description

The command retrieves the spec block parameters for the tributaries specified by the input parameter *aid*.

Example input/output

```
RTRV-ALL:LT-UNITE:1-1-#-#-ALL-ALL-ALL:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-3-1-1:crslpbkstat=NO,tmonmd=NMON,tribsdhpn=Default,
ptca=Default,neppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-6,...,
ptrcrfmt=1,dexcthr=-3,ptrc=00,sdsfmode=POISSON″

   ″1-1-#-#-3-1-2:crslpbkstat=NO,tmonmd=NMON,tribsonetpn=Default,
ptca=Default,neppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-6,...,
ptrcrfmt=1,dexcthr=-3,ptrc=00,sdsfmode=POISSON″

   ″1-1-#-#-3-1-3:crslpbkstat=NO,tmonmd=NMON,tribsonetpn=Default,
```

```
ptca=Default,neppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-6,...,
ptrcrfmt=1,dexcthr=-3,ptrc=00,sdsfmode=POISSON"
                            .
                            .
                            .
                            .
                            .
                            .
                            .
   "1-1-#-#-3-1-192:crslpbkstat=NO,tmonmd=NMON,tribsonetpn=Default,
ptca=Default,neppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-6,...,
ptrcrfmt=1,dexcthr=-3,ptrc=00,sdsfmode=POISSON"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-ALL** command.

Related TL1 messages

- "RTRV-VC3: Retrieve VC3 tributary" (p. 3-3069)
- "RTRV-STS1: Retrieve STS1 tributary" (p. 3-2946)

References

GR-199-CORE (RTRV-*rr*)

RTRV-ALM: Retrieve alarm

The **RTRV-ALM** command messages can be initiated by the user to retrieve all active alarms with properties defined by command parameters of level critical, major or minor from a NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ALM**[-*modifier*]:*tid*::*ctag*::[*ntfcncde*],[*condtype*],[*srveff*],,,;

The last three trailing commas are used to specify that the optional parameters *locn*, *dirn*, *tmper* as defined in GR-833 are omitted since these parameters are not supported as input parameters.

Input parameters

Command parameters

1. **modifier** – Message Modifier
   This parameter defines for which AID types the conditions will be retrieved. If the value 'ALL' is used, the specified parameter has no impact on the set of retrieved alarms delivered in the retrieve reply.
   Type: Alphanumeric value set.
   Range: ALL. This requests all alarms.
   The modifier can be omitted. In this case the ALL modifier will be set implicit.
2. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **ntfcncde** – Notification Code
   This is the alarm severity (alarm level) for which the current alarms are requested.
   Type: Alphanumeric value set.
   Range: ALL, requests all alarms.
   ntfcncde can be omitted. In this case ALL will be set implicit.

5. **condtype** – Condition Type
   This parameter identifies the type of alarm indication being reported.
   Type: Alphanumeric value set.
   Range: ALL, requests all alarms.
   condtype can be omitted. In this case ALL will be set implicit.
6. **srveff** – Service Effect
   This parameter identifies the service affect of alarms being reported.
   Type: Alphanumeric value set.
   Range: ALL, requests all alarms.
   srveff can be omitted. In this case ALL will be set implicit.

Output format

The following example shows the successful completion of the **RTRV-ALM** command by the NE:

```
   sid date time
M  ctag COMPLD
   ″aid,aidtype:ntfcncde,condtype,srveff,ocrdat,ocrtm:conddescr″
   (0 or more lines of the above)
;
```

If no alarms are raised only COMPLD will be returned.

The alarms are reported one line per condition, similar to the autonomous messages conveyed by REPT ALM.

Output parameters

Command parameters

1. **aid** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: All defined AIDs are allowed. It should be a valid AID for the specified *modifier*. Refer to the definition of AIDs.
2. **aidtype** – Access Identifier Type
   This parameter defines the AID type to which the alarm belongs.
   Type: Alphanumeric value set.
   Range:
   - COM: Common. An alarm/event that applies to the whole NE system; for example, a processing error.
   - EQPT: An equipment-related alarm/event.
   - SYST: An alarm/event related to system timing.
   - OUTT: An alarm/event related to output timing.

- STM1E: A facility-related alarm/event for synchronous transport mode 1 (electrical).
- STM1: A facility-related alarm/event for synchronous transport mode 1.
- STM4: A facility-related alarm/event for synchronous transport mode 4.
- STM16: A facility-related alarm/event for synchronous transport mode 16.
- STM16T: A facility-related alarm/event for fully transparent synchronous transport mode 16.
- STM64: A facility-related alarm/event for synchronous transport mode 64.
- STM256: A facility-related alarm/event for synchronous transport mode 256.
- ODU1: A facility-related alarm/event for optical data units of level 1
- OC3: A facility-related alarm/event at the optical carrier signal level 3.
- OC12: A facility-related alarm/event at the optical carrier signal level 12.
- OC48: A facility-related alarm/event at the optical carrier signal level 48.
- OC48T: A facility-related alarm/event at the fully transparent optical carrier signal level 48.
- OC192: A facility-related alarm/event at the optical carrier signal level 192
- OC768: A facility-related alarm/event at the optical carrier signal level 768
- T3: A facility-related alarm/event at the DS3 level
- TMAT3: A facility-related alarm/event at the DS3 level related to TransMUX function
- EC1: A facility-related alarm/event at the EC1 level
- 1GE: A facility-related alarm/event for 1 Gigabit Ethernet.
- 1GEPL: A facility-related alarm/event for 1 Gigabit Ethernet Private Line.
- 10GEPL: A facility-related alarm/event for 10 Gigabit Ethernet Private Line.
- VC3: A facility-related alarm/event for virtual container 3.
- VCG: A facility-related alarm/event for virtual concatenation group.
- VC4: A facility-related alarm/event for virtual container 4
- VC44C: A facility-related alarm/event for a signal concatenated at 4 times the VC4 rate.
- VC416C: A facility-related alarm/event for a signal concatenated at 16 times the VC4 rate.
- VC464C: A facility-related alarm/event for a signal concatenated at 64 times the VC4 rate.
- STS1: A facility-related alarm/event at synchronous transport signal level 1.
- TMASTS1: A facility-related alarm/event at synchronous transport signal level 1 related to TransMUX function.
- STS3C: A facility-related alarm/event at synchronous transport signal level 3.

- STS12C: A facility-related alarm/event at synchronous transport signal level 12.
- STS48C: A facility-related alarm/event at synchronous transport signal level 48.
- STS192C: A facility-related alarm/event at synchronous transport signal level 192.
- HOVC4 : A facility-related alarm/event for virtual container 4 higher-order path termination.
- HOSTS1: A facility-related alarm/event at synchronous transport signal level 1 higher-order path termination.
- LOVC3: A facility-related alarm/event for virtual container 3 (lower-order path).
- LOVC12: A facility-related alarm/event for virtual container 12
- LOVT1: A facility-related alarm/event at transport level VT1.5.
- TMSVT: A facility-related alarm/event at transport level VT1.5 related to TransMUX function.
- DS1: A facility-related alarm/event at transport level DS1.
- LINK: A signaling communication link related alarm/event for ONNS

Refer to "Alarms, events, ASAPs" (p. 6-4) for the relation of alarm / entity / *aidtype*.

3. **`conddescr`** – Description of the Condition
   This is a textual Description of the Condition.
   Type: inner string of the format:
   \″<category>, <entity>, <description>, <additional text>\″ or
   \″<category>, <entity>, <description>, <additional text>-cleared\″ (this form is used in REPT ALM/EVT for cleared alarms)
   Range:
   - category: is the category
   - entity: is the entity for which the alarm is reported
   - description: is the Alarm description, the string is up to 32 characters
   - additional text: additional text.

   If there is no additional text, then the trailing comma may be omitted.
4. **`condtype`** – Condition Type
   This parameter identifies the probable cause (type of alarm indication) being reported.
   Type: Alphanumeric value set.
   Range: All defined probable causes.
5. **`ntfcncde`** – Notification Code
   This is the alarm severity (alarm level) or if the alarm is cleared.
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)

- MN – Minor (ITU / Telcordia)
- CL – Cleared.

6. **`srveff`** – Service Effect
   This parameter identifies the service affect of alarms being reported. It presents the "best guess" from a local system point of view.
   Type: Alphanumeric value set.
   Range:
   - SA: Service-Affecting alarm condition
   - NSA: Non-Service-Affecting alarm condition.
7. **`ocrdat`** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
8. **`ocrtm`** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.

Detailed behavior description

RTRV-ALM retrieves all active alarms of level CR, MJ, MN.

Any alarm reported by an autonomous REPT ALM message which is active when a RTRV-ALM message is received, is included in the RTRV-ALM response message. Alarms of level NA are reported by RTRV-COND and not by RTRV-ALM.

RTRV-ALM is an on-demand version of the autonomous REPT ALM. The on-demand command only report alarms that are still set when the command is executed.

The **`REPT-ALM`** command does not generate a **`REPT DBCHG`** message.

```
RTRV-ALM:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-4-1,STM16:CR,LOS,SA,01-01,07-29-13:\″Communication-Transport, STM16
      port, Loss Of Signal\″″
   ″1-1-#-#-1-3,STM4:MJ,LOF,SA,12-31,03-50-32:\″Communication-Transport, STM4 port,
      Loss of Frame\″″
;
```

Example input/output

The following example shows the successful completion of the **RTRV-ALM** command by the NE if no *modifier* is specified and all alarms will be retrieved for the whole NE:

The following example shows the successful completion of the **RTRV-ALM** command by the NE if the alarms for a specified *aid* will be retrieved:

```
RTRV-ALM:LT-UNITE-123:1-1-#-#-4-1:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-4-1,STM16:CR,LOS,SA,01-01,07-29-13:\″Communication-Transport, STM16
      port, Loss Of Signal\″″
;
```

The following example shows the successful completion of the **RTRV-ALM** command by the NE if no alarms are raised on the NE:

```
RTRV-ALM:LT-UNITE-123:1-1-#-#-3-1:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ALM** command.

Related TL1 messages

- “ALW-MSG: Allow message” (p. 3-33)
- “ENT-ASAP-PROF: Enter alarm severity assignment profile” (p. 3-1446)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “RTRV-ASAP-PROF: Retrieve ASAP profile” (p. 3-2249)
- “INH-MSG: Inhibit message” (p. 3-2021)
- “REPT ALM: Report alarm” (p. 3-2096)
- “REPT-ALM-ENV: Report alarm environment” (p. 3-2101)
- “REPT EVT: Report event” (p. 3-2121)
- “RTRV-ALM-ENV: Retrieve alarm environment” (p. 3-2238)
- “RTRV-COND: Retrieve condition” (p. 3-2285)
- “RTRV-HDR: Retrieve header” (p. 3-2417)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: NE and Transport Surveillance Messages

RTRV-ALM-ENV: Retrieve alarm environment

The **RTRV-ALM-ENV** command retrieves active environmental alarms of the severity CR, MJ, MN (Critical, Major, or Minor (ITU/Telcordia)).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ALM-ENV**:*tid*:[*aid*]:*ctag*::[*ptfcncde*],[*almtype*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDI port entities for which an alarm condition or the configuration is being reported. If misc_inall is specified the information for all entities will be retrieved.
   Type: AID.
   Range: Miscellaneous_in_discreteAID.
   Default: misc_inall (if the aid is omitted).
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **ntfcncde** – Notification Code
   This is the alarm severity (alarm level) for which the current alarms are requested.
   Type: Alphanumeric value set.
   Range: ALL, requests all alarms.
   ntfcncde can be omitted. In this case ALL will be set implicit.
5. **almtype** – Alarm Type
   This parameter identifies the type of alarm indication being reported.
   Type: Alphanumeric value set.
   Range: MISC (default), an event that applies to the miscellaneous discrete outputs inputs.

Output format

The following example shows the successful completion of the **RTRV-ALM-ENV** command by the NE:

```
   sid date time
M  ctag COMPLD
   "aid:ntfcncde,almtype,ocrdat,ocrtm[,almmsg]"
   (0 or more lines of the above)
;
```

If no alarms are raised only COMPLD will be returned.

Output parameters

Command parameters

1. **aid** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: Miscellaneous_in_discreteAID. The AID misc_inall must not appear.
2. **almtype** – Alarm Type
   This parameter identifies the type of alarm indication being reported.
   Type: Alphanumeric value set.
   Range: MISC (default), an event that applies to the miscellaneous discrete outputs inputs.
3. **ntfcncde** – Notification Code
   This is the alarm severity (alarm level) for which the current alarms are requested.
   Type: Alphanumeric value set.
   Range: ALL, requests all alarms.
   ntfcncde can be omitted. In this case ALL will be set implicit.
4. **ocrdat** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
5. **ocrtm** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
6. **almmsg** – Alarm Message
   A string describing the alarm of the environmental point.
   Note: This string is configurable by the **SET-ATTR-ENV** command.
   Type: text string.
   Range: upper and lower case, up to 32 characters.
   Default: Miscellaneous Discrete.

Detailed behavior description

Any environmental alarm reported by an autonomous REPT ALM ENV message, which is active when a RTRV-ALM-ENV message is received, is included in the RTRV-ALM-ENV response message.

Environmental alarms of severity NA (warning (ITU) / not alarmed (Telcordia)) are not included in a RTRV-ALM-ENV response. They are included in the RTRV-COND response.

Example input/output

The following example shows the successful completion of the **RTRV-ALM-ENV** command by the NE:

```
RTRV-ALM-ENV:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″MISC_IN7:MJ,MISC,01-01,07-49-23,\″Miscellaneous Discrete #7\″″
   ″MISC_IN5:MN,MISC,01-01,06-32-25,\″stairway 5 temperature\″″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ALM-ENV** command.

Related TL1 messages

- “ALW-MSG: Allow message” (p. 3-33)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “INH-MSG: Inhibit message” (p. 3-2021)
- “REPT ALM: Report alarm” (p. 3-2096)
- “REPT-ALM-ENV: Report alarm environment” (p. 3-2101)
- “REPT EVT: Report event” (p. 3-2121)
- “RTRV-ALM: Retrieve alarm” (p. 3-2231)
- “RTRV-COND: Retrieve condition” (p. 3-2285)
- “SET-ATTR-ENV: Set attribute environment” (p. 3-3137).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

RTRV-AO: Retrieve autonomous output

The **RTRV-AO** command is used to retrieve copies of past **REPT DBCHG** messages for users in the memory administration user class.

The copies of **REPT DBCHG** messages available for retrieval include those actually reported to the user as well as those that may not have reached the user due to external (for example, DCN failures) or internal (for example, inhibited autonomous TL1 message reporting) conditions.

If messages are inhibited with the **INH-MSG** command, the internal buffering of autonomous messages is unaffected.

The system saves a copy of each **REPT DBCHG** message for users in the memory administration user class (*uidclass* = MEMADMIN). Those copies share one *atag* counter for all users of the memory administration user class. Message is saved even if no user of the memory administration user class is actually logged into the system.

This is in contrast to the autonomous messages which are sent to users of user classes different from memory administration, where the *atag* counter is available per user session.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

User Class: The user must belong to the user class *uidclass* = MEMADMIN (see "RTRV-USER: Retrieve user" (p. 3-3059)).

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-AO**:*tid*::*ctag*[:::*spec_block*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **atagseq** – ATAG Sequence
   This parameter specifies one or more *atags* for which the **REPT DBCHG** messages is retrieved from the log.

Examples of ranges that are allowed:

- *atagseq*=10 → Retrieve *atag* message 010
- *atagseq*=10&20 → Retrieve *atag* message 010, 020
- *atagseq*=10&&20 → Retrieve *atag* message 010-020
- *atagseq*=999&&1 → Retrieve *atag* message 999, 000, 001.

If the value is omitted, all messages are retrieved, regardless of *atag*.
Type: (range of) numeric string.
Examples of ranges that are allowed:

- *atagseq*=10 Retrieve **REPT DBCHG** message with *atag* 010
- *atagseq*=10&20 Retrieve **REPT DBCHG** message with *atag* s 010, 020
- *atagseq*=10&&20 Retrieve **REPT DBCHG** message with *atag* s 010-020
- *atagseq*=999&&1 Retrieve **REPT DBCHG** message with *atag* s 999, 000, 001.

Range: 0 .. 999
Default: *null*, if the value is omitted, all messages are retrieved, regardless of *atag*.

2. **msgtype** – Message Type
Retrieve only atag messages of this specific type.
Type: Alphanumeric value set.
Range: DBCHG: Retrieve messages reported by **REPT DBCHG**.
Default: *null*, if the value is omitted, only DBCHG messages are retrieved.

Output format

If there are autonomous messages to report, the following response is returned:

```
   sid date time
M  ctag COMPLD
/*
complete autonomous messages without the termination indicator (”;”)
*/
;
```

Sorting

If more than one **REPT DBCHG** message is returned, the output is sorted by ascending time order. If the system clock is reset, the output is sorted in the order in which the messages were stored into the log.

For example, if the system originally sent messages 998 through 001, and the user requests those messages, the *atag* numbers in the output will be in the order: 998, 999, 000, 001.

If the *atag* numbers have wrapped and the most recent entry is *atag* 040, and if the user requests *atagseq*=38&&43, the *atag* order in the output will be 041, 042, 043, 038, 039, 040.

Detailed behavior description

The system returns all **REPT DBCHG** messages matching the *atag* range of the input parameter *atagseq* to the user.

Update affected entities: Execution of this command does not affect other entities in the system.

Example input/output

The following is an example of the successful execution of a RTRV-AO command:

```
RTRV-AO:LT-UNITE:12:123456;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
/*
   LT-UNITE 00-11-01 07:50:00
A  012 REPT DBCHG
   ″DATE=00-11-01,TIME=07-49-58:ED-NE:::tunstat=ENABLE″
*/
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command Error Response section. The error responses listed there also apply to the **RTRV-AO** command.

Validation rule: valid *atagseq*

If the NE receives a RTRV-AO command with an invalid *atagseq* value the command is rejected with

Error code: `IDNV – Input Data Not Valid, invalid` *`atagseg`*

Validation rule: user class

If the NE receives a RTRV-AO command from a user that is not in the memory administration user class, i.e. *uidclass* = MEMADMIN, the command is rejected with

Error code: `PICC – Privilege, Illegal Command Code, invalid user class`

Related TL1 messages

- “RTRV-LOG-NTFCN: Retrieve notification log” (p. 3-2464)
- “REPT DBCHG: Report data base change” (p. 3-2118)

References

GR-199-CORE

RTRV-ASAP-ASGNMT: Retrieve ASAP assignment

The **RTRV-ASAP-ASGNMT** command retrieves a list of AIDs that identifies all users of an ASAP of a specific type and a specific name.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ASAP-ASGNMT**:*tid*:*aid*:*ctag*::*pftype*,*pfname*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   Identifies the entity for which and all contained entities the ASAP assignments will be retrieved. Only for a shelf and all its contained entities all ASAP assignments can be retrieved.
   Type: AID.
   Range: ShelfAID.
4. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch

- OTN: OTN ports.
- PATH: Path MSA/NIM related
- PATHTERM: Path Termination
- PORT: SDH/SONET ports
- SCN: SCN
- SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

5. **`pfname`** – ASAP Profile Name
   This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
   Range: length 1-24 characters.
   For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Output format

The following example shows the successful completion of the **`RTRV-ASAP-ASGNMT`** command by the NE:

```
   sid date time
M  ctag COMPLD
   ″aid_output:pftype,pfname″
   (0 or more lines)
;
```

Output parameters

Command parameters

1. **`aid`**_output – Access Identifier
   Access Identifier of an entity which has an ASAP pointer which points to an ASAP of *pftype*, *pfname*.
   Type: AID.
   Range: All AIDs of entities which contain a ASAP pointer. For entities see in the section ″Entity – ASAP Relation″.
   Default: None.
2. **`pftype`** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring

- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

3. **pfname** – ASAP Profile Name
   This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by *pftype*. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
   Range: length 1-24 characters.
   For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Detailed behavior description

The *RTRV-ASAP-ASGNMT* command does not generate a REPT DBCHG message.

Behavior: No matching profile name

If the system receives a command with a *pfname* that does not exist, then the command will successfully complete with an empty reply list. Thus no data will be returned. Note: A wrong pftype is handled by the general error response IDNV (RTRV-HDR).

Behavior with DEFAULT profile

If the system receives a command with pfname = DEFAULT, then the command will successfully complete with an empty reply list.

Example input/output

The following example shows the successful completion of the **RTRV-ASAP-ASGNMT** command by the NE:

```
RTRV-ASAP-ASGNMT:LT-UNITE-123:1-1:123456::SONET,GTE;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-2:SONET,GTE″
   ″1-1-#-#-1-3:SONET,GTE″
   ″1-1-#-#-4-1:SONET,GTE″
   ″1-1-#-#-4-1:SONET,GTE″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ASAP-ASGNMT** command.

Related TL1 messages

- “DLT-ASAP-PROF: Delete alarm severity assignment profile” (p. 3-134)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “RTRV-ASAP-PROF: Retrieve ASAP profile” (p. 3-2249)
- “ENT-ASAP-PROF: Enter alarm severity assignment profile” (p. 3-1446).

RTRV-ASAP-PROF: Retrieve ASAP profile

The **RTRV-ASAP-PROF** command retrieves information about one or more alarm severity assignment profile(s).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ASAP-PROF**:*tid*::*ctag*::*pftype*[,*pfname*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **pftype** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen ("-") is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN

- SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
- TIMING: System Timing
- TRANSPARENCY: Fully Transparent Service

Default Value: There is no default value. The *pftype* does not change.

4. **`pfname`** – ASAP Profile Name
   This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen ("-") is allowed in the name additionally.
   Range: length 1-24 characters.
   For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Output format

The following example shows the successful completion of the **`RTRV-ASAP-PROF`** command by the NE if the *pfname* parameter is omitted:

```
   sid date time
M  ctag COMPLD
   ″pftype=pftype,name=DEFAULT″
   ″pftype=pftype,name=pfname″
   ...
   ″pftype=pftype,name=pfname″
;
```

There may be 0 or more *pfname* lines, where each line gives the name of an ASAP of type *pftype*. Even if ENT-ASAP-PROF has never been used to create a profile for this pftype, RTRV-ASAP-PROF will still return one profile: DEFAULT.

Normal completion response if *pfname* is specified

The following example shows the successful completion of the **`RTRV-ASAP-PROF`** command by the NE if the *pfname* parameter is specified:

```
   sid date time
M  ctag COMPLD
   ″pftype=pftype,name=pfname″
   ″spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Important! Instead of returning an error message the **`RTRV-ASAP-PROF`** command returns success with an empty list when the user tries to retrieve non-existing profiles.

Output parameters

Command parameters

1. **`pftype`** – Profile Type
   This parameter indicates the type of the ASAP (ASAP type).
   Type: Alphanumeric value set. A hyphen (“-”) is allowed for the alphanumeric values additionally.
   Range:
   - DCN: Data Communications Network
   - DS1: DS1 asynchronous signals
   - DS3: DS3 asynchronous ports
   - ENET: Ethernet
   - ENV: Environmental (Miscellaneous Discrete)
   - EQUIP-PROT: Equipment with Protection State Dependence
   - EQUIP-UNPROT: Equipment unprotected
   - MS-SPRing-BLSR: MS-SPRing and Bilateral Switched Ring protection switch
   - MSP-APS: Automatic protection switch
   - OTN: OTN ports.
   - PATH: Path MSA/NIM related
   - PATHTERM: Path Termination
   - PORT: SDH/SONET ports
   - SCN: SCN
   - SNCP-UPSR: Sub-Network Connection Protection and Unidirectional Path Switched Ring
   - TIMING: System Timing
   - TRANSPARENCY: Fully Transparent Service

   Default Value: There is no default value. The *pftype* does not change.
2. **`pfname`** – ASAP Profile Name
   This parameter indicates the name of the ASAP. The ASAP name distinguishes ASAPs of the same ASAP type specified by pftype. That means the same name can be used for several ASAP types.
   Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
   Range: length 1-24 characters.
   For predefined ASAPs is the value of *pfname* ″DEFAULT″, but it is no default name in general.

Spec block parameters

1. Alarm Severity Profile parameters
   The spec block contains ASAP parameters. Each ASAP parameter defines for a probable cause (*condtype*) the assigned severity which will be reported when the alarm appears. If for one specific probable cause a service affecting (sa) or non service affecting (nsa) alarm can be reported then for each the sa-alarm and nsa-alarm an ASAP parameter exists.
   The severity indicates the urgency of an alarm. The severity of a sa-alarm must be more urgent that of a nsa-alarm for the same *condtype*.
   The description of the ASAP parameters, the relation to the profile types and the default values are described in: “Alarms, events, ASAPs” (p. 6-4).
   All parameters have the same type and value range.
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)
   - MN – Minor (ITU / Telcordia)
   - NA – Warning (ITU) / Not-Alarmed (Telcordia)
   - NR – Non-Alarmed (ITU) / Not-Reported (Telcordia).

   Example: Concrete ASAP parameters are *sa_eqpt* and *nsa_eqpt*. Both refers to the probable cause (*condtype*) EQPT and ASAP type EQUIPPROT. *sa_eqpt* contains the assigned severity for the service affecting alarm of probable cause EQPT and *nsa_eqpt* for the non service affecting alarm. *sa_eqpt* can have the concrete value CR and *nsa_eqpt* MN.

Detailed behavior description

The sysem returns

- All defined profiles of a specific profile type, if only the profile type is specified
- All profile parameters of a specific profile, if profile type and profile name is specified.

If no profile with the specified name exists, the response is empty.

The **RTRV-ASAP-PROF** command does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of the **RTRV-ASAP-PROF** command by the NE if *pfname* is omitted:

```
RTRV-ASAP-PROF:LT-UNITE-123::123456::ENV;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ”Pftype=ENV,Name=DEFAULT”
```

```
   ”Pftype=ENV,Name=GTE”
;
```

The following example shows the successful completion of the **RTRV-ASAP-PROF** command by the NE if *pfname* and *pftype* is specified:

```
RTRV-ASAP-PROF:LT-UNITE-123::123456::ENET,GTE;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ”Pftype=ENET,Name=GTE”
   ”sa_vcg_loa=MJ,nsa_vcg_loa=MN,sa_vcg_ssf=MJ,nsa_vcg_ssf=MN,
sa_vcg_plm=MN,nsa_vcg_plm=NA,sa_gfp_lof=CR,nsa_gfp_lof=MJ,sa_lan_los=CR,nsa_lan_
      los=MJ,sa_lan_anm=MN,nsa_lan_anm=NA”
;
```

Error responses

The **RTRV-ASAP-PROF** command does not generate any special error responses. The general error responses defined in "RTRV-HDR: Retrieve header" (p. 3-2417) are also valid for this command.

Related TL1 messages

- "DLT-ASAP-PROF: Delete alarm severity assignment profile" (p. 3-134)
- "ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
- "ENT-ASAP-PROF: Enter alarm severity assignment profile" (p. 3-1446)
- "RTRV-ASAP-ASGNMT: Retrieve ASAP assignment" (p. 3-2244)

RTRV-ATTR-CONT: Retrieve attribute control

The **RTRV-ATTR-CONT** command can be initiated by users to retrieve the configuration associated with an miscellaneous output control (MDO port / external discrete control). The assigned descriptions are used for information purposes when a user seeks to operate or release external miscellaneous discrete controls.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ATTR-CONT**:*tid*:[*aid*]:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDO port entities on which the command operates. If misc_outall is specified the configuration of all MDO’s will be retrieved.
   Type: AID
   Range: Miscellaneous_out_discreteAID
   Default: misc_outall (if the *aid* is omitted).
3. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

The following example shows the successful completion of the **RTRV-ATTR-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
   ”aid:contmsg,durmode,initcstat,[asgncond]”
   ...
   ”aid:contmsg,durmode,initcstat,[asgncond]”
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   This is the address of the MDO port entity for which the configuration is retrieved or on which an operation is performed.
   Type: AID.
   Range: Miscellaneous_out_discreteAID. misc_outall must not be reported or used.
2. **contmsg** – Control Message
   This is the name associated to a MDO port (environmental control). It is a description which can be retrieved by the management system.
   Type: inner string.
   Range: string up to 26 characters, any character except ″ and \, upper and lower cases considered. A empty string is allowed. The value would be \″\″.
   Default: MDO port #n. n is the number of the port 1 … 8.
3. **durmode** – Duration Mode
   This parameter defines in which mode the device connected to the MDO is expected to be operated. It is a fixed value because this feature will be not supported by the system. This parameter is in for compatibility reason. The system operates only continuous that means if the MDO is operated it is switched on and it is released it is switched off.
   Type: Alphanumeric value set.
   Range: CONTS, continuous duration.
4. **initcstat** – Initial Control State
   This parameter defines the state the MDO must have in order to put the connected device in the ″off″-state. For UNITE it is a fixed value because this feature will be not supported. This parameter is in for compatibility reason.
   Type: Alphanumeric value set.
   Range: RLS, release.
5. **asgncond** – Assigned Condition Type
   This parameter identifies the probable cause (type of alarm indication) which is assigned to an MDO port (external control). If one or more alarm(s) of this probable cause appears then the MDO port (external control) will be switched on.
   Type: Alphanumeric value set. Additionally a “-” will be allowed (not the first character).
   Range: All defined probable causes and UNDEF: if no probable cause is assigned.
   Default: UNDEF.

Detailed behavior description

The **RTRV-ATTR-CONT** command does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of the **RTRV-ATTR-CONT** command by the NE:

```
RTRV-ATTR-CONT:LT-UNITE-123:MISC_OUT3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ”MISC_OUT3:\”fan 3\”,CONTS,RLS,FANF”
;
```

The following example shows the successful completion of the **RTRV-ATTR-CONT** command by the NE if the aid is omitted:

```
RTRV-ATTR-CONT:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ”MISC_OUT2:\”sprinkler 2\”,CONTS,RLS”
   ”MISC_OUT3:\”fan 3\”,CONTS,OPER,FANF”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ATTR-ENV** command.

Related TL1 messages

- “OPR-EXT-CONT: Operate external control” (p. 3-2032)
- “RLS-EXT-CONT: Release external control” (p. 3-2190)
- “RTRV-ALM: Retrieve alarm” (p. 3-2231)
- “RTRV-ACTL-CONT: Retrieve actual condition types” (p. 3-2219)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

RTRV-ATTR-ENV: Retrieve attribute environment

The **RTRV-ATTR-ENV** command is used to retrieve the alarm message, name and assigned ASAP of a miscellaneous discrete input (MDI/environmental point).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-ALM-ENV**:*tid*:[*aid*]:*ctag*;

Input parameters

Command parameters

- **tid** – Target Identifier
  See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
- **aid** – Access Identifier
  This is the address of the MDI port entities for which an alarm condition or the configuration is being reported. If misc_inall is specified the information for all entities will be retrieved.
  Type: AID.
  Range: Miscellaneous_in_discreteAID.
  Default: misc_inall (if the aid is omitted).
- **ctag** – Correlation Tag
  See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

The following example shows the successful completion of the **RTRV-ATTR-ENV** command by the NE:

```
   sid date time
M  ctag COMPLD
   "aid:,,envmsg,envname:pfname,envinv"
   (more lines for default AID)
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: Miscellaneous_in_discreteAID. The AID misc_inall must not appear.
2. **`envinv`** – Environmental Alarm Inversion
   This parameter indicates for the MDI (environmental point) whether an alarm is generated if the port is on or if the port is off.
   Type: Alphanumeric value set.
   Range:
   - YES: The alarm will be inverted
   - NO: The alarm will be not inverted.
3. **`envmsg`** – Environment Message
   This is the condition description which is associated with the MDI port (environmental point). This description will be sent with the alarm generated from the MDI port.
   Type: inner string.
   Range: a string up to 32 characters, any character except ″ and \, upper and lower cases are considered.
   Default: Miscellaneous Discrete #<N> with N = 1 .. 8 for the different MDI ports.
4. **`envname`** – Environmental Point Name
   This is the name associated to a MDI port (environmental point). It is a description which can be retrieved by the management system. It is not to be mixed up with the envmsg parameter. The envname is just a name of the MDI port and not reported via the condition description of an alarm.
   Type: inner string
   Range: a string up to 26 characters, any character except ″ and \, upper and lower cases considered.
   Default: MDI port #n; n is the number of the port 1 … 8.
5. **`pfname`** – Profile Name
   This is the name of the ASAP that is assigned to the MDI port (environmental point). The ASAP must be of the type ENV.
   Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
   Range: string up to 24 characters.
   Default: DEFAULT.

Detailed behavior description

The **`RTRV-ATTR-ENV`** command does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of the **RTRV-ATTR-ENV** command by the NE:

```
RTRV-ATTR-ENV:LT-UNITE-123:misc_in3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″MISC_IN3:,,\″smoke detected on detector 3\″,\″smoke detector 3\″:ENV_CR,NO″
;
```

The following example shows the successful completion of the **RTRV-ATTR-ENV** command by the NE if all MDI's will be retrieved:

```
RTRV-ATTR-ENV:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″MISC_IN1:,,\″pump on\″:ENV_NA,NO″
   ″MISC_IN2:,,\″smoke detected on detector 1\″,\″smoke detector 1\″:ENV_CR,NO″
   ″MISC_IN3:,,\″smoke detected on detector 2\″,\″smoke detector 2\″:ENV_CR,NO″
   ″MISC_IN4:,,\″smoke detected on detector 3\″,\″smoke detector 3\::ENV_CR,NO″
   ″MISC_IN5:,,\″smoke detected on detector 4\″,\″smoke detector 4\″:ENV_CR,NO″
   ″MISC_IN6:,,\″fan on\″,\″fan\″:ENV_NA,YES″
   ″MISC_IN7:,,\″open door\″,\″door\″:DEFAULT,YES″
   ″MISC_IN8:,,\″open door\″,\″door 2\″:DEFAULT,YES″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-ATTR-ENV** command.

Related TL1 messages

- "REPT-ALM-ENV: Report alarm environment" (p. 3-2101)
- "RTRV-ALM-ENV: Retrieve alarm environment" (p. 3-2238)
- "SET-ATTR-ENV: Set attribute environment" (p. 3-3137).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

RTRV-ATTR-MSG: Retrieve attribute message

The **RTRV-ATTR-MSG** command is used to retrieve the current status of the autonomous message reporting, which can be controlled via INH-MSG and ALW-MSG.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: yes

Input format

Command syntax: **RTRV-ATTR-MSG***:tid::ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the RTRV-ATTR-MSG request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″sessions:msgstate″
   . . .
;
```

For each value of parameter *sessions* one line is reported.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

1. **`sessions`** – Affected sessions
   This parameter specifies for which sessions the *msgstate* is applied.
   Type: Alphanumeric value set.
   Range: CURRENT: affected session is the current session of the user who executed the command.
   Default: CURRENT
2. **`msgstate`** – Message status
   The current state of autonomous message reporting for *sessions*.
   Type: Alphanumeric value set.
   Range:
   - ON: autonomous reporting is enabled for *sessions*
   - OFF: autonomous reporting is disabled for *sessions*.

   Default: ON

Detailed behavior description

The command retrieves the current status of the autonomous message reporting for all values of parameter *sessions*. Control of this state is via the commands INH-MSG and ALW-MSG.

When a user logs into the system via ACT-USER it is recommended to retrieve the status of autonomous reporting to check whether reporting is OFF for *sessions* = ALL.

Example input/output

The following is an example of the **`RTRV-ATTR-MSG`** command:

```
RTRV-ATTR-MSG:LT-UNITE::123456;

   LT-UNITE 00-11-15 08:00:00
M  123456 COMPLD
   ″ALL:ON″
   ″CURRENT:OFF″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`INH-MSG`** command.

Related TL1 messages

- “ALW-MSG: Allow message” (p. 3-33)
- “INH-MSG: Inhibit message” (p. 3-2021)

References

GR-833-CORE

RTRV-AUDLOG: Retrieve audit result log for ONNS connection data

Overview

The **RTRV-AUDLOG**-*mod* command is used to request information about the last audit result on ONNS data that is stored. Note that this command does not perform the audit operation. See AUD-mod TL1 command for the details of audit operation. The modifier of the RTRV-AUDLOG-mod command indicates the scope of data the audit operation was performed on.

RTRV-AUDLOG-LOCAL

RTRV-AUDLOG-LOCAL shall retrieve the last audit result based on comparison of ONNS data with the master data from local host NE. Mismatched cross-connects and port objects between two sets of data will be displayed..

RTRV-AUDLOG-PATHNBR

RTRV-AUDLOG-PATHNBR shall retrieve the last audit result based on comparison of ONNS path data with neighboring ONNS nodes.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command Syntax of RTRV-AUDLOG-LOCAL:
RTRV-AUDLOG-LOCAL:*srcTid:srcAid:ctag;*

Command Syntax of RTRV-AUDLOG-PATHNBR:
RTRV-AUDLOG-PATHNBR:*srcTid:srcAid:ctag;*

Input parameters

Command parameters

1. **srcTid**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **ctag**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").

Output format

If the RTRV-AUDLOG-LOCAL request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″ExtraPortIDs=1portID,2portID,...,N1portID″

   ″MissingPortIDs=1portID,2portID,...,N2portID″

   ″ExtraNNXcons=1connPtID1,-X-,1connPtID2,1bandwidth,1pathID
   :2connPtID1,-X-,2connPtID2,2bandwidth,2pathID
   :...
   :N3connPtID1,-X-,N3connPtID2,N3bandwidth,N3pathID″

   ″MissingNNXcons=1connPtID1,-X-,1connPtID2,1bandwidth
   :2connPtID1,-X-,2connPtID2,2bandwidth
   :...
   :N4connPtID1,-X-,N4connPtID2,N4bandwidth″

   ″ExtraNonNNXcons=1connPtID1,-X-,1connPtID2,1bandwidth
   :...
   :N5connPtID1,-X-,N5connPtID2,N5bandwidth″
```

```
   ″MissingNonNNXcons=1connPtID1,-X-,1connPtID2,1bandwidth
   :...
   :N6connPtID1,-X-,N6connPtID2,N6bandwidth″

   ″MultipleNNXcons=1connPtID1,-X-,1connPtID2,1connPtID3,-X-
    ,1connPtID4, 1bandwidth, 1pathID
   :...
   : N7connPtID1,-X-,N7connPtID2,N7connPtID3,-X-
    ,N7connPtID4,N7bandwidth, N7pathID″
;
```

The output shows any mismatched port objects and mismatched cross connect information in comparing NN data and host NE data.

For mismatched port objects, the cases may occur are as follows:

- port object exists only in NN data, but not in host NE data (ExtraPortIDs)
- port object exists only in host NE data, but not in NN.

The cases in the response represent the mismatched port objects or mismatched cross connect information as follows:

- ExtraPortIDs – port object exists only in NN data, but not in host NE data
- MissingPortIDs – pport object exists only in host NE data, but not in NN data
- ExtraNNXcons – NN owned cross-connect exists only in NN data, but not in host NE
- MissingNNXcons – NN owned cross-connect exists only in host NE data, but not in NN data
- ExtraNonNNXcons – non-NN owned cross-connect exists only in NN data, but not in host NE
- MissingNonNNXcons – non-NN owned cross-connect exists only in host NE data, but not in NN data.

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line. See Aud-LOCAL for the details of audit operation.

If the RTRV-AUDLOG-PATHNBR request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″MismatchedPathIDs=1connPtID1,-X-
      ,1connPtID2,1bandwidth,1localpathID,1nbrnodeID,1nbrpathID
   :2onnPtID1,-X-,2connPtID2,2bandwidth,2localpathID,2nbrnodeId,2nbrpathID
   :...
   :NconnPtID1,-X-,NconnPtID2,Nbandwidth,NlocalpathID,NnbrnodeID,NnbrpathID″
   ″NotReachedNeighborNodes=1nbrnodeId,2nbrnodeID,...,NnbrnodeID″
```

;

The output shall show any mismatched path data in comparing path information at a local NN with its neighboring nodes.

The cases in the response represent the mismatched path information as follows:

- When all the parameters have valid values – local node and neighboring node have different pathIDs for the same path.
- When the parameter for pathID at neighboring node (nbrpathID) is empty – local node has a cross-connect that is connected to a neighboring node, but the neighboring node does not have the corresponding cross-connect or path information.
- When the parameter for pathID at local node (localpathID) is empty – neighboring node has a cross-connect that is connected to a local node, but the local node does not have the corresponding cross-connect or path information.

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

All neighbor nodes which are not reachable are be listed. For these neighbor nodes is was not possible to retrieve information for comparison of the ONNS path data.

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *conpletion code* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

1. **`pathID`**
   The PathID of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0..255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65111
2. **`localpathID`**
   The PathID of the local node of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0..255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65535

3. **`nbrpathID`**
   The PathID of the neighbor node of the specific path to be reported.
   Type: character string of length up to 20
   Range: nodeID-Index. *nodeID*: four dot separated numeric strings: each string 0..255 *Index*: numeric string 0-65535.
   The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203-65111

4. **`nbrnodeID`**
   ONNS node Network Access identifier (nodeID) of the neighbor node. *nodeID* is not changable and may be different from Signaling network IP address.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0..255.

5. **`bandwidth`** – Requested bandwidth / connection *signalRate*
   ONNS bandwidth is equal to or greater than the minimum cross connect rate (minXCrate) of the port type. Note: Only SONET switching levels in enumeration. For actual cross-connections ONNS adapts to SDH formats automatically if the ingress port is SDH.
   Type: Value set.
   Values: STS1, STS3, STS12, STS48, STS192.

6. **`portID`**
   portID of port object discrepancy between NN data and NE data. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # . ONNS assumes a bidirectional port pair.
   Type: dash-separated list of 5 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#). This range is valid for protecting connections in M:N connection groups

7. **`connPtID1`**
   ONNS connection point identifier (connPtID1) for the ingress RX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #..
   Type: dash-separated list of 5 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
   Range: Bay/Frame ## (1), Shelf # (1), Slot/Pack # #, Port # #, Tributary ##. Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
   Range: Slot/Pack#*, Port#*, Tributary#*
   Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional

cross-connections.
Range: Slot/Back#@, Port#@, Tributary#@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

8. **connPtID2**
ONNS connection point identifier (connPtID2) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##. Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
Range: Slot/Pack#*, Port#*, Tributary#*
Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
Range: Slot/Back#@, Port#@, Tributary#@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

9. **connPtID3**
ONNS connection point identifier (connPtID3) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##. Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
Range: Slot/Pack#*, Port#*, Tributary#*
Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional

cross-connections.
Range: Slot/Back#@, Port#@, Tributary#@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

10. **connPtID4**
ONNS connection point identifier (connPtID4) for the egress TX connPt of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##. Default tributary AID used for reporting the not existing xcons in the A-node or Z-node for a path associated to a protecting NN connection of an M:N connection group.
Range: Slot/Pack#*, Port#*, Tributary#*
Value * for Slot/Pack, Port and Tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter connPtA) is allocated by one or more LO traditional cross-connections.
Range: Slot/Back#@, Port#@, Tributary#@
Value @ for Slot/Pack, Port and Tributary indicates that the corresponding HO input and/or output tributary specified by connPtA is involved as EDGE-prt tributary in an OUC.

Example input/output

The following example shows a successful completion of a RTRV-AUDLOG command for local ONNS data audit operation by the NE with no data is out-of-sync:

```
RTRV-AUDLOG-LOCAL:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″ExtraPortIDs=NONE″
   ″MissingPortIDs=NONE″
   ″ExtraNNXcons=NONE″
   ″MissingNNXcons=NONE″
   ″ExtraNonNNXcons=NONE″
   ″MissingNonNNXcons=NONE″
   ″MultipleNNXcons=NONE″
;
```

The following example shows a successful completion of a RTRV-AUDLOG command for local ONNS data audit operation by the NE:

```
RTRV-AUDLOG-LOCAL:LT-ONNS-789:snn:123456;
```

```
   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″ExtraPortIDs=1-1-20-3-5″
   ″MissingPortIDs=NONE″
   ″ExtraNNXcons=1-6-5-3-1,-X-,1-2-4-3-1,sts48,29.13.196.203-43216
   :1-1-4-4-1,-X-,1-2-1-3-1,sts12,29.13.18.23-12345″
   ″MissingNNXcons=1-1-8-10-2,-X-,1-1-3-3-4,sts48″
   ″ExtraNonNNXcons=NONE″
   ″MissingNonNNXcons=NONE″
   ″MultipleNNXcons=NONE″
;
```

The symbol (-X-) indicates that a cross-connection is present between the *connPtIDs* on an output line.

The following example shows a successful completion of a RTRV-AUDLOG command for ONNS path neighbor data audit operation by the NE with no data is out-of-sync:

```
RTRV-AUDLOG-PATHNBR:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″MismatchedPathIDs=NONE″
   ″NotReachedNeighborNodes=NONE″
;
```

The following example shows a successful completion of a RTRV-AUDLOG command for ONNS path neighbor data audit operation by the NE:

```
RTRV-AUDLOG-PATHNBR:LT-ONNS-789:snn:123456;

   LT-ONNS-789 2003-01-01 08:00:00
M  123456 COMPLD
   ″MismatchedPathIDs=1-1-4-4-1,-X-,1-2-1-3-1,sts12,29.13.18.23-
      00000,168.16.29.111,135.14.196.203-32165
   :1-1-7-4-2,-X-,1-1-3-5-1,sts48,,29.13.18.23,135.14.196.109-12164″
   ″NotReachedNeighborNodes=29.12.13.1,29.12.13.2″
;
```

The symbol (-X-) indicates that a cross-connection is present between the connPtIDs on an output line.

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ED-SYNCDATA: Synchronization of ONNS data" (p. 3-1198)
- "AUD: Perform audit data" (p. 3-46)

Possible issues when widely using unidirectional ONNS connections

When in an ONNS network unidirectional ONNS connections are widely used, the following misbehavior may occur:

1. Local audit erroneously reports extra NN cross-connections: **AUD-LOCAL** and **RTRV-AUDLOG** may erroneously report extra NN cross-connections, if two unidirectional ONNS connections of differing bandwidth are traversing a node in opposite directions, but with connecting the same timeslots, for example, one connection from timeslot A to timeslot B and the other one from timeslot B to timeslot A.
2. Free bandwidth generally advertised on an INNI link, although no more bandwidth for bidirectional ONNS connections available: Setup of a bidirectional ONNS connection via an INNI link may fail, although for this link enough free bandwidth is generally advertised for example, via **RTRV-NTDR**. This is the case, if the corresponding link is nearly totally filled with a major part being unidirectional connections. The advertised free bandwidth may then refer to differing timeslots for both directions, thus not allowing bidirectional connections to be setup.

RTRV-BANNER

The **RTRV-BANNER** command can be initiated by users to retrieve the user-provisioned banner for the System.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: No.

Input format

Command syntax: **RTRV-BANNER**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

After receiving the **RTRV-BANNER** command, the following normal completion response is returned::

```
   sid date time
M  ctag COMPLD
   ":::spec_block"
    ...
    < 1 line for each Spec Block parameter to be reported >
    ...
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Spec block parameters

1. **`line1`** – Line 1 of the Customer Provisionable Banner.
   Setting line1 to an empty string will suppress the output of the Customer Provisionable Banner in the response to the **`ACT-USER`** command.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.
2. **`line 2`** … line 11 – Line 2 (…Line 11) of the Customer Provisionable Banner.
   Type: text string
   Range: 2 … 62 characters, case-sensitive. The beginning and ending double quotes are included.
   Default: none. The initial value is an empty string.

Detailed behavior description

The **`RTRV-BANNER`** command shall return one line for each provisioned line of the Customer Provisionable Banner. A line of the Customer Provisionable Banner is provisioned, if the corresponding spec block parameter (line1 … line 11) is not an empty string.

Example input/output

The following example shows the successful completion of the **`RTRV-BANNER`** command by the network element:

```
RTRV-BANNER:LT-UNITE-123::123456;

   LT-UNITE-123 04-12-23 08:00:00
M  123456 COMPLD
″:::LINE1=\″NOTICE: THIS IS A PRIVATE COMPUTER SYSTEM.\″″
″:::LINE2=\″UNAUTHORIZED ACCESS OR USE MAY LEAD TO PROSECUTION.\″″
;
```

Related TL1 messages

- "ACT-USER: Activate user" (p. 3-24)
- "ENT-BANNER" (p. 3-1452).

RTRV-CKT: Retrieve circuit

The **RTRV-CKT** retrieves circuits which use a specified BLSR/MS-SPRing ring channel. The command is only functional in rings with automatic ring discovery enabled.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-CKT**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies a tributary of a BLSR/MS-SPRING port for which the circuit information is requested to be retrieved.
   Type: AID
   Range: GenTribAID (not ranged)
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Output format

After receiving the **RTRV-CKT** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "rid"
   "ntid,nid,src,dest,[cctype],[rate][:spec_block]"
   ...
   ...
   "ntid,nid,src,dest,[cctype],[rate][:spec_block]"
;
```

Output parameters

Command parameters

1. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the *rid* to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings
   Range: 1 to 15 characters
2. **`ntid`** – Node Target Identifier
   This parameter identifies the TID of the node on which the reported cross-connection exists. This is the currently active name of the NE to which the command is addressed. The *tid* is a case insensitive character string.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: 1 to 20 characters
3. **`nid`** – Node ID
   This parameter specifies the node ID of the NE within the ring as used by the MS-SPRing/BLSR APS protocol. It shall correspond to the first TID in the *linkid* block.
   Type: Numeric string.
   Range: 0 … 15
4. **`src`** – Source
   This parameter specifies the tributary which is the source of the cross-connection or displays an error identifier for the input of a missing xconn.
   Type: Value set.
   Range: 20 characters

   The allowed values are:

   - GenTribAID: a tributary AID for a local xconn
   - REM-<AID>TS#: This value is reported as the *aid* in an MS SPRING / BLSR worker port on a remote node. <AID> is replaced by the received worker port AID of a remote node (note: The syntax of the remote AID will be not checked; it is only displayed) and # identifies the timeslot number on the interface. (e.g.: REM-m1-1-TS1)
   - REM-PPORT-TS#: This value is reported as the *aid* in an MS SPRING / BLSR protection port on a remote node.

     # identifies the timeslot number on the interface. (e.g.: REM-PPORT-TS5)

- REMAID: This value is reported as the input aid of an add/drop cross-connection at a remote node.
- PIF (Path Integrity Failure): This value is used to identify the input aid of a missing connection.

5. **`dest`** – Destination
   This parameter specifies the destination which is the sink of the cross-connection or displays an error identifier for the output of a missing xconn.
   Type: Value set.
   Range: 20 characters
   The allowed values are:
   - GenTribAID: a tributary AID for a local xconn
   - REM-<AID>TS#: This value is reported as the *aid* in an MS SPRING / BLSR worker port on a remote node. <AID> is replaced by the received worker port AID of a remote node (note: The AID can be not in the UNITE syntax; it is only displayed) and # identifies the timeslot number on the interface.
   - REM-PPORT-TS#: This value is reported as the *aid* in an MS SPRING / BLSR protection port on a remote node.
     # identifies the timeslot number on the interface. (e.g.: REM-PPORT-TS5)
   - REMAID: This value is reported as the input aid of an add/drop cross-connection at a remote node.
   - PIF (Path Integrity Failure): This value is used to identify the input aid of a missing connection.
6. **`cctype`** – Cross Connection Type
   This parameter specifies the type of the cross-connection:
   Type: Alphanumeric value string
   Range: UNIDIR (default)
7. **`rate`** – Cross Connection Rate
   This parameter indicates the cross-connection rate. For a detailed description please refer to the input parameter *modifier*.
   Type: Alphanumeric value string
   Range: LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C, LOVT1, STS1, STS3, STS12, STS48, STS192
   Conversion Cross-Connections:
   - AU3TU3 (HO VC3/STS1 input to LO VC3 output)
   - TU3AU3 (LO VC3 input to HO VC3/STS1 output)

Spec block parameters

1. **exreestbl** – Extra Traffic Channel Reestablishment
   The parameter exreestbl shall indicate, whether the extra traffic circuit is a candidate for reestablishment wrt ring switching or span switching for the addressed span. The parameter is only supported for protection groups in the local node with protection switch mode TOPEX. For remote nodes or protection groups other than TOPEX the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - NONE: no candidate for reestablishment
   - RING: a candidate for reestablishment via ring switching
   - SPAN: a candidate for reestablishment via span switching
   - RINGSPAN: a candidate for reestablishment via ring and span switching
   - NA: Not Applicable (for service traffic channels)
2. **loca** – Location-a
   This parameter specifies the TID of the node at which service is added to the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections.
   The *loca* parameter value is an empty string for ADD and non-BLSR connections. A 2way cross-connection is considered to be a DROP if the *out_aid* is a non BLSR tributary.The figure “2-way scenario” (p. 3-2322)in RTRV-CRS section (the 3 NEs should be seen as part of an BLSR) depicts the 2way scenario.
   Inter ring cross-connections with collapsed node: Note that beside the above described THROUGH case there can be also another situation where both *loca* and *locz* are different from the empty string: If two rings are interconnected in a collapsed node then *loca* refers to the (add-node of the) ring where the signal is dropped from, and *locz* refers to the (drop-node of the) ring where the signal is added to. In such cases the cross-connection cannot be provisioned as bidirectional, thus the information is still unambiguous.
   A note to the *loca2* parameter: Although *loca2* (Location-a-2) is a parameter which is used in the **ENT-CRS** command, *loca2* is not retrieved explicitly. This is due to the fact that for path-protected cross-connections – and only for those this *loca2* is applicable – the retrieval will be on a leg-basis and not on a leg-pair basis. In other words: If a path-protected cross-connection is retrieved two entries will be sent, one for the working leg (incl. the *loca* and *locz* parameter) and one for the protection leg (incl. the *loca* parameter and the same *locz* parameter, and in this case the *loca* returned is the *loca2* as entered when establishing the path-protected cross-connection).
   The parameter loca is also required for EXTRA TRAFFIC connections. (In contrast to incoming references from other commands to this whole chapter it is exclusively applicable for the command **RTRV-CRS**.). For extra traffic the value will be generated automatically and thus it is only retrievable.
   For information about the automatic generation of *loca* refer to the parameter *sqfa*.

For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides. If there is no such function in an upstream ring node then *loca* is the TID of the node where the related HO circuit is first added to the ring. The figure "Lower-order cross-connections" (p. 3-2322) in RTRV-CRS section shows an example of a LO path through as part of a ring.
The LO path is added to/dropped from the ring in NE-1 and NE-2. 2-way cross-connections are considered to be either an Add or a Drop connection depending whether the input or the output tributary is at the ring port. In NE-3 there is a HO through connection, which contains the LO path. In NE-3 there is no HO path termination. In NE-4 there is a LO through connection. Consequently, there are HO path terminations in NE-4.

The fact that *loca* refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same *loca* value.
- Any LO through-connection will be treated as add/drop. *loca* will identify the node of the through connection, and not the node where the LO service is added to the ring.

At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
Type: TID or empty-string
Range: Please refer to the TID definition in the RTRV-HDR description
Default: empty string: ″″
Note: In output direction the empty string appears as \″\″.

3. **locz** – Location-z
This parameter identifies specifies the TID of the node at which service is dropped from the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections. The *locz* parameter value is an empty string for DROP and non-BLSR connections.
A 2-way cross-connection is considered to be a ADD if the *in_aid* is a non BLSR tributary. The figure "2-way scenario" (p. 3-2322)in RTRV-CRS section (the 3 NEs should be seen as part of an BLSR) depicts the 2way scenario.
locz is also required for EXTRA TRAFFIC connections. For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of *locz* refer to the parameter *sqfz*.
For lower-order cross-connections the parameter *locz* is the TID of that ring node in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides. If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring. Refer to the figure "Lower-order cross-connections" (p. 3-2322)in the RTRV-CRS section for further explanation.

The fact that *locz* refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same *locz* value.
- Any LO through-connection will be treated as add/drop. *locz* will identify the node of the through connection, and not the node where the LO service is dropped from the ring.

At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.
Type: TID or empty-string.
Range: Please refer to the TID definition in the RTRV-HDR description
Default: empty string: ""
Note: In output direction the empty string appears as \"\".

4. **`rappl`** – Ring Application
The ring application is as given by the contents of the local squelch map in the direction of circuit tracing. This parameter depicts the cross-connection application in the remote nodes and the local node for the direction of the circuit tracing.
Type: Alphanumeric value set.
Range: ADD, DROP, THRU, TSI-THRU, DROPMUL, ADDSEL, DROPMUL-PROT, ADDSEL-PROT

5. **`srvcrest`** – Service Channel Restoration
The parameter indicates, whether the service traffic circuit is a candidate for restoration wrt. ring switching or span switching for the addressed span. The parameter is only supported for protection groups in the local node with protection switch mode TOPEX. For remote nodes or protection groups other than TOPEX the parameter is omitted.
Type: Alphanumeric value set.
Range:
   - NONE: no candidate for reestablishment
   - RING: a candidate for reestablishment via ring switching
   - SPAN: a candidate for reestablishment via span switching
   - RINGSPAN: a candidate for reestablishment via ring and span switching
   - NA: Not Applicable (for service traffic channels)

Detailed behavior description

No cross-connection

If there is no cross-connection at the logical input or output channel/tributary in the local node, the command shall complete, but no circuits are reported.

Specified tributary does not exist

If the tributary specified in the command does not exist, then the command shall complete with an empty response.

Only unidirectional cross-connections

A bidirectional (2-way) cross-connection is handled as two unidirectional (1-way) cross-connections.

Order of circuits and direction of tracing

If the STS-N channel / tributary as input of the Retrieve Circuit command is source and / or destination of more than one local *xconn*, the local *xconns* is sorted as follows:

- a local unidirectional / point_to_multipoint xconn with the given tributary as destination point, tracing direction of the associated circuit is the egressing direction of the port with the given tributary
- a local unidirectional / point_to_multipoint xconn with the given tributary as source point, tracing direction of the associated circuit is the ingressing direction of the port with the given tributary

In this sequence the circuits associated with the local xconns are listed in the reponse of the command.

Order of xconns in a circuit

The remote and local xconns of a circuit is sorted beginning with the xconn of the source node and ending with the xconn of the destination node whereas source and destination nodes are determined by the direction of tracing. In case that there are more than one local xconn legs contributing to the circuit they is listed consecutively starting with the through part.

For the local NE there are several output lines, one for each cross-connection leg of the circuit. For remote NEs there is only one line reported. The *rappl* values ADDSEL or DROPMUL indicate that there is more than one cross-connection leg in the romote NE.

Not complete circuits

If a circuit discontinuity is detected, xconn listing shall start/end with the node with the discontinuity using the error indicator PIF for that side continuity was lost.

For the xconn in the node with the discontinuity only the following parameters are returned:

- *ntid* of the node
- *nid* of the node
- *in_aid*
- *out_aid*

Not complete circuits

If a circuit discontinuity is detected, xconn listing shall start/end with the node with the discontinuity using the error indicator for that side continuity was lost.

For the xconn in the node with the discontinuity only the following parameters are returned:

- *ntid* of the node
- *nid* of the node
- *in_aid*
- *out_aid*

Wrap around circuits

If a wrap around circuit is detected for the local xconn, xconn listing shall start with the local xconn in the direction as given by the STS-N channel/tributary. The xconn input tributary is the error identifier PIF. xconn listing shall comprise xconns in all nodes of the ring and shall end in the neighbor node of the local node with the error identifier PIF for the output tributary.

Overlap circuits

If an overlap circuit is detected for the local xconn, xconn listing shall either:

- start in the node with the ADD xconn of the circuit and end in the neighbor node with the error identifier PIF for the output tributary after one complete cycle or
- start in the neighbor node of that with the DROP xconn with an error identifier PIF for the input tributary and end in the node with the DROP xconn after one complete cycle.

Example input/output

Normal output response – no circuit

RTRV-CKT for no circuit:

```
RTRV-CKT:unite:1-1-#-#-1-7:123456;
   unite 04-12-24 08:00:00
M  123456 COMPLD
;
```

Circuit in non-TOP+EX Ring with only UNITE NEs

RTRV-CKT for the THRU NE: The unidirectional circuit is added in UNITE0, through connected in UNITE1 and dropped in UNITE2. RTRV-CKT is done on UNITE1

```
RTRV-CKT:UNITE1:1-1-#-#-22-1-1:123;
IP 123
<


   UNITE1 04-12-28 00:22:15
M  123 COMPLD
   ″ring0″
   ″UNITE0,1,REMAID,REM-1-1-#-#-22-1-TS1,UNIDIR,STS1:locz=UNITE2,rappl=ADD″
```

```
   "UNITE1,0,1-1-#-#-22-1-1,1-1-#-#-23-1-1,UNIDIR,STS1:loca=UNITE0,locz=UNITE2,
      rappl=THRU"
   "UNITE2,2,REM-1-1-#-#-24-1-TS1,REMAID,UNIDIR,STS1:loca=UNITE0,rappl=DROP"
;
```

RTRV-CKT for the ADD NE: The unidirectional circuit is added in UNITE0 and dropped in UNITE1. RTRV-CKT is done on UNITE0.

```
RTRV-CKT:UNITE0:1-1-#-#-01-1-1:123;
IP 123
<


   UNITE0 04-12-28 00:22:15
M  123 COMPLD
   "ring0"
   "UNITE0,1,1-1-#-#-01-1-1,1-1-#-#-22-1-1,UNIDIR,STS1:locz=UNITE1,rappl=ADD"
   "UNITE1,0,REM-1-1-#-#-24-1-TS1,REMAID,UNIDIR,STS1:loca=UNITE0,rappl=DROP"
;
```

Circuit in non-TOP+EX Ring with mixed NEs

RTRV-CKT for the THRU NE: The unidirectional circuit is added in UNITE0, through connected in UNITE1 and dropped in DMX. The drop node (DMX) is a non-UNITE NE. RTRV-CKT is done on UNITE1:

```
RTRV-CKT:UNITE1:1-1-#-#-22-1-1:123;
IP 123
<


   UNITE1 04-12-28 00:22:15
M  123 COMPLD
   "ring0"
   "UNITE0,1,REMAID,REM-1-1-#-#-22-1-TS1,UNIDIR,STS1:locz=DMX,rappl=ADD"
   "UNITE1,0,1-1-#-#-22-1-1,1-1-#-#-23-1-1,UNIDIR,STS1:loca=UNITE0,locz=DMX,
      rappl=THRU"
   "DMX,2,REM-m2-1-TS1,REMAID,UNIDIR,STS1:loca=UNITE0, rappl=DROP"
;
```

Circuit in TOP+EX Ring with only UNITE NEs

RTRV-CKT for the ADD NE: The unidirectional circuit is added in UNITE0 and dropped in UNITE1. RTRV-CKT is done on UNITE0:

```
RTRV-CKT:UNITE0:1-1-#-#-01-1-1:123;
IP 123
<


   UNITE0 04-12-28 00:22:15
```

```
M  123 COMPLD
   ″ring0″
   ″UNITE0,1,1-1-#-#-01-1-1,1-1-#-#-22-1-
      1,UNIDIR,STS1:exreestbl=NA,locz=UNITE1,rappl=ADD,srvcrest=RINGSPAN″
   ″UNITE1,0,REM-1-1-#-#-24-1-TS1,REMAID,UNIDIR,STS1:loca=UNITE0,rappl=DROP″
;
```

Not complete circuit in non-TOP+EX Ring with only UNITE NEs

RTRV-CKT for the THRU NE: The unidirectional circuit is added in UNITE0, through connected in UNITE1 and ends in UNITE2 without drop or through connection. RTRV-CKT is done on UNITE1:

```
RTRV-CKT:UNITE1:1-1-#-#-22-1-1:123;
IP 123
<


   UNITE1 04-12-28 00:22:15
M  123 COMPLD
   ″ring0″
   ″UNITE0,1,REMAID,REM-1-1-#-#-22-1-TS1,UNIDIR,STS1:rappl=ADD″
   ″UNITE1,0,1-1-#-#-22-1-1,1-1-#-#-23-1-1,UNIDIR,STS1:rappl=THRU″
   ″UNITE2,2,REM-1-1-#-#-24-1-TS1,PIF
;
```

Error responses

Validation rule: no BLSR / MS-SPRING tributary

The Retrieve Circuit command is denied if a tributary of a non BLSR / MS-SPRING port is addressed.

Error code: `IDNV – Input, Data Not Valid, no BLSR / MS SPRING tributary`

Validation rule: no automatic squelch map calculation

The Retrieve Circuit command is denied if automatic squelch map calculation is not enabled for the ring.

Error code: `IDNV – Input, Data Not Valid, no Auto. Squelch Map Calculation`

Validation rule: Invalid ring map

The Retrieve Circuit command is denied if no valid ring map exists or the ring circuit map discovery is not yet completed.

Error code:

`SNVS – Status, Not in Valid State, ring circuit map discovery not yet completed`

Related TL1 messages

- "RTRV-MAP-RING: Retrieve map ring" (p. 3-2532)
- "ED-MAP-RING: Edit map ring" (p. 3-436).
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
- "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
- "RTRV-MAP-NEIGHBOR: Retrieve map neighbor" (p. 3-2522)
- "RTRV-MAP-NETWORK: Retrieve network map" (p. 3-2526)
- "SET-MAP-RING: Set map ring" (p. 3-3142)

References

GR-199-CORE.

TID information in RTRV-CKT is not updated for a TID change of a DMX BLSR node

In a mixed 1665 Data Multiplexer Extend (DMX) / 1675 LambdaUnite MSS BLSR configuration with auto squelch map calculation enabled, a TID change of a DMX node will not be reflected in the response of the TL1 Retrieve Circuit command (RTRV-CKT). The reason is that a TID change of a DMX node does not cause a reboot as for 1675 LambdaUnite MSS and that thus no ring circuit map redistribution is triggered.

Workaround

To correct the **RTRV-CKT** result, toggle automatic ring discovery (*ard*) and auto squelch map calculation (*asmc*) to disabled and to enabled again. Now the **RTRV-CKT** response will show the updated TID information.

RTRV-COND: Retrieve condition

RTRV-COND command messages can be initiated by the user to retrieve all active alarms of level NA – Not Alarmed (Telcordia) / Warning (ITU).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-COND**[-*modifier*]:*tid*::*ctag*::[*condtype*],,,;

The last three trailing commas are used to specify that the optional parameters *locn*, *dirn*, *tmper* as defined in GR-833 are omitted since these parameters are not supported as input parameters.

Input parameters

Command parameters

1. **modifier** – Message Modifier
This parameter defines for which AID types the conditions will be retrieved. If the value ALL is used, the specified parameter has no impact on the set of retrieved alarms delivered in the retrieve reply.
Type: Alphanumeric value set.
Range: ALL, this requests all alarms.
The modifier can be omitted. In this case the ALL modifier will be set implicit.
2. **tid** – Target Identifier
See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **ctag** – Correlation Tag
See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **condtype** – Condition Type
This parameter identifies the type of alarm indication being reported.
Type: Alphanumeric value set.
Range: ALL, requests all alarms.
condtype can be omitted. In this case ALL will be set implicit.

Output format

Normal completion response of successful session termination.

The following example shows the successful completion of the **RTRV-COND** command by the NE:

```
   sid date time
M  ctag COMPLD
   ”aid,aidtype:ntfcncde,condtype,srveff,ocrdat,ocrtm
   [,,,,conddescr]”
   (0 or more lines of the above)
;
```

If no alarms are raised only COMPLD will be returned.

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID
   Range: All defined AIDs are allowed. It should be a valid AID for the specified *modifier*. Refer to the definition of AIDs.
2. **`aidtype`** – Access Identifier Type
   This parameter defines the AID type to which the alarm belongs.
   Type: Alphanumeric value set.
   Range:
   - COM: Common. An alarm/event that applies to the whole NE system; for example, a processing error.
   - EQPT: An equipment-related alarm/event.
   - SYST: An alarm/event related to system timing.
   - OUTT: An alarm/event related to output timing.
   - STM1E: A facility-related alarm/event for synchronous transport mode 1 (electrical).
   - STM1: A facility-related alarm/event for synchronous transport mode 1.
   - STM4: A facility-related alarm/event for synchronous transport mode 4.
   - STM16: A facility-related alarm/event for synchronous transport mode 16.
   - STM16T: A facility-related alarm/event for fully transparent synchronous transport mode 16.
   - STM64: A facility-related alarm/event for synchronous transport mode 64.
   - STM256: A facility-related alarm/event for synchronous transport mode 256.
   - ODU1: A facility-related alarm/event for optical data units of level 1
   - OC3: A facility-related alarm/event at the optical carrier signal level 3.
   - OC12: A facility-related alarm/event at the optical carrier signal level 12.
   - OC48: A facility-related alarm/event at the optical carrier signal level 48.
   - OC48T: A facility-related alarm/event at the fully transparent optical carrier signal level 48.

- OC192: A facility-related alarm/event at the optical carrier signal level 192
- OC768: A facility-related alarm/event at the optical carrier signal level 768
- T3: A facility-related alarm/event at the DS3 level
- TMAT3: A facility-related alarm/event at the DS3 level related to TransMUX function
- EC1: A facility-related alarm/event at the EC1 level
- 1GE: A facility-related alarm/event for 1 Gigabit Ethernet.
- 1GEPL: A facility-related alarm/event for 1 Gigabit Ethernet Private Line.
- 10GEPL: A facility-related alarm/event for 10 Gigabit Ethernet Private Line.
- VC3: A facility-related alarm/event for virtual container 3.
- VCG: A facility-related alarm/event for virtual concatenation group.
- VC4: A facility-related alarm/event for virtual container 4
- VC44C: A facility-related alarm/event for a signal concatenated at 4 times the VC4 rate.
- VC416C: A facility-related alarm/event for a signal concatenated at 16 times the VC4 rate.
- VC464C: A facility-related alarm/event for a signal concatenated at 64 times the VC4 rate.
- STS1: A facility-related alarm/event at synchronous transport signal level 1.
- TMASTS1: A facility-related alarm/event at synchronous transport signal level 1 related to TransMUX function.
- STS3C: A facility-related alarm/event at synchronous transport signal level 3.
- STS12C: A facility-related alarm/event at synchronous transport signal level 12.
- STS48C: A facility-related alarm/event at synchronous transport signal level 48.
- STS192C: A facility-related alarm/event at synchronous transport signal level 192.
- HOVC4 : A facility-related alarm/event for virtual container 4 higher-order path termination.
- HOSTS1: A facility-related alarm/event at synchronous transport signal level 1 higher-order path termination.
- LOVC3: A facility-related alarm/event for virtual container 3 (lower-order path).
- LOVC12: A facility-related alarm/event for virtual container 12
- LOVT1: A facility-related alarm/event at transport level VT1.5.
- TMSVT: A facility-related alarm/event at transport level VT1.5 related to TransMUX function.

- DS1: A facility-related alarm/event at transport level DS1.
- LINK: A signaling communication link related alarm/event for ONNS

Refer to "Alarms, events, ASAPs" (p. 6-4) for the relation of alarm / entity / *aidtype*.

3. **conddescr** – Description of the Condition

   For events which belong to an ASAP profile:

   - This is a textual description of the condition.
   - Type: inner string of the format: <category>, <entity>, <description>, <additional text>, or <category>, <entity>, <description>, <additional text>-cleared (this form is used in REPT ALM/EVT for cleared alarms).
   - Range: category: is the category; entity: is the entity for which the alarm is reported; description: is the Alarm description, the string is up to 32 characters; additional text = additional text. If there is no additional text, then the trailing comma may be omitted.
   - The general format is: category, entity, description, additional text. If there is no additional text, then the trailing comma may be omitted.
   - Type: inner string
   - Range for STM<N>, STM<N>T, OC<N>T and OC<N>: Quality of service, rr port, TCA; where rr is equal to the port rate
   - Range for ODU<N>: Quality of service, rr path, TCA; where rr is equal to the ODU rate (e.g., ODU1)
   - Range for not terminated VC12, VC3, VC4, VC44C, VC416C, VC464C, VT1, STS1, STS3C, STS12C, STS48C and STS192C signals: Quality of service, rr path, TCA; where rr is equal to the signal rate.
   - Range for terminated VC4 and STS1 signals: Quality of service, rr path, TCA; where rr is equal to the Constituent Signal (CS) rate.
   - Range for GE port: Quality of service, {ge_rate}GE, TCA where ge_rate is the signal rate of the GE port.
   - Range for VCG port: Quality of service, VCG, TCA
   - Range for DS3 port: Quality of service, T3 port, TCA
   - Range for TMAT3 port: Quality of service, T3 port, TCA
   - Range for TMASTS1 path termination: Quality of service, {***rr***} path, TCA, where ***rr*** is equal to STS1 rate
   - Range for not terminated DS1 tributary: Quality of service, DS1 path, TCA

   For ABN conditions see definition of *conddescr*:

   - The general format is: category, description
   - Type: inner string
   - Range, description for a protection group AID: Abnormal condition, forced switch active; Abnormal condition, lockout active

- Range, description for a timing reference AID: Abnormal condition, operate in hold-over; Abnormal condition, operate in free run mode
- Range, description for a port AID: Abnormal condition, facility loopback active
- Range, description for a logical tributary AID: Abnormal condition, cross connect loopback active; Abnormal condition, test access session active
- Range, description for a system AID: – Abnormal condition, System operates in maint. mode

4. **`condtype`** – Condition Type

   For events which belong to an ASAP profile:

   - This parameter identifies the probable cause (type of alarm indication) being reported.
   - Type: Alphanumeric value set.
   - Range: all defined probable causes.
   - This parameter identifies the type of event indication being reported.
   - Type: inner string with the Format: T-<montype>
   - Range for SDH signals

     The following values are for all *STM<N> signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-MS-N-ES, T-MS-N-SES, T-MS-N-BBE, T-MS-N-UAS, T-MS-F-ES, T-MS-F-SES, T-MS-F-BBE, T-MS-F-UAS

     The following values are for all *STM<N>T signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-RS-N-ES-EGR, T-RS-N-SES-EGR, T-RS-N-BBE-EGR, T-RS-N-UAS-EGR

     The following values also are for *STM64 and STM256 signals*: T-MS-N-FECC

     The following values are for *VC3, VC4, VC44C, VC416C and VC464C tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, cVC-F-BBE, T-VC-F-UAS, T-PJCPOS, T-PJCNEG

     The following values are for *VC12 tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, T-VC-F-BBE, T-VC-F-UAS
   - Range for SONET signals

     The following values are for all *OC<N> and EC1 signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVL, T-ESL, T-SESL, T-UASL, T-CV-LFE, T-ES-LFE, T-SES-LFE, T-UAS-LFE

     The following values are for all *OC<N>T signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVS-EGR, T-ESS-EGR, T-SESS-EGR, T-SEFS-S-EGR

     The following values are also for *OC192 and OC768 signals*: T-FECC-L

     The following values are for STS1, STS3C, STS12C, STS48C and STS192C tributaries: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE, PPJC-PDet, NPJC-PDet, NPJC-PDet, NPJC-PGen

     The following values are for *TMASTS1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP, T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

The following values are for *VT1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

- Range for DS3/DS1 signals

  The following values are for *DS3 signals with T3 modifier*:

  T-CVL, T-ESL, T-ESA-L, T-ESB-L, T-SESL, T-CVP, T-ESP, T-ESA-P, T-ESB-P, T-SESP, T-UASP, T-SAS-P, T-CV-PFE, T-ES-PFE, T-ESA-PFE, T-ESB-PFE, T-SES-PFE, T-UAS-PFE, T-SAS-PFE, T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS3 signals with TMAT3 modifier*:

  T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS1 signals*:

  T-CVP-EGR, T-ESP-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR

- Range for other signals

  The following values are for all *ODU<N> signals*:

  T-ODU-N-ES, T-ODU-N-SES, T-ODU-N-BBE, T-ODU-N-UAS, T-ODU-F-ES, T-ODU-F-SES, T-ODU-F-BBE, T-ODU-F-UAS

  For *ENET signals*:

  T-PDE, T-EDFE, T-EDFC

Default: None.

For ABN conditions see definition of *condtype* :

- This parameter identifies the type of event indication being reported.
- Type: inner string with the Format: ABN-<abntype>
- Range: ABN-FORCESW, Forced switch active. ABN-LOCKOUT, Lockout active. ABN-HLDOVRSYNC, Holdover synchronization mode. ABN-FREERUNSYNC, Free-run synchronization mode. ABN-LPBKPORT, Facility Loopback. ABN-LPBKXCON, Cross Connect Loopback. ABN-MAINTENANCE, System operates in maintenance mode. ABN-TESTACC, Test access session has been setup

5. **`ntfcncde`** – Notification Code
   This is the alarm level (severity).
   Type: Alphanumeric value set.
   Range: NA – Warning (ITU) / Not-Alarmed (Telcordia)
6. **`ocrdat`** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.

7. **`ocrtm`** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
8. **`srveff`** – Service Affect
   This parameter identifies the service affect of alarms being reported.
   Type: Alphanumeric value set.
   Range:
   - SA – Service affecting
   - NSA – Non service affecting.
9. **`tmper`** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.

Detailed behavior description

RTRV-COND command messages can be initiated by the user to retrieve all active alarms of level NA – Not Alarmed (Telcordia) / Warning (ITU) from a NE.

RTRV-COND is closely related to REPT EVT which autonomously reports when an persistent alarm of notification code NA either sets or clears or an transient event occurs (like reporting of protection switch states). RTRV-COND retrieves the REPT EVT events that are still set. protection switching events that are reported by REPT EVT are not reported by RTRV-COND (for example, the lockout of a timing reference is not reported by RTRV-COND).

In addition to reporting all active alarms of level NA previously reported by REPT EVT, RTRV-COND retrieves those reported by other autonomous messages.

The **`RTRV-COND`** command does not generate a **`REPT DBCHG`** message.

The **`RTRV-COND`** command will also report the active TCA-events, for port or trib entities, which work in SDH mode and the user has set the TCA mode tcamd of the entity to TR-RTR. By switching to TR-RTR the TCA events are reported as a standing condition, thus the system creates a raise and clear event. If TR-only is selected the TCA events are handled as a transient condition and therefore never reported by RTRV-COND. The SONET standard supports only TR-only mode.

Example input/output

The following example shows the successful completion of the **RTRV-COND** command by the NE.

```
RTRV-COND:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   "1-1-#-#-6-1,OC48:NA,AIS-L,NSA,01-01,07-59-59,,,,\"Communication-Transport, OC48
      port, Alarm Indication Signal\""
   "MISC_IN3,COM:NA,MISC,NSA,01-01,07-20-15,,,,\"Environment, Misc Discrete,door
      open\""
;
```

The following example shows the successful completion of the **RTRV-COND** command by the NE if an *aid* is specified:

```
RTRV-COND:LT-UNITE-123:MISC_IN3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   "MISC_IN3,COM:NA,MISC,NSA,01-01,07-20-15,,,,\"Environment, Misc Discrete,door
      open\""
;
```

The following example shows the successful completion of the **RTRV-COND** command by the NE if no conditions are raised:

```
RTRV-COND:LT-UNITE-123:MISC_IN3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ALM** command.

Related TL1 messages

- “ALW-MSG: Allow message” (p. 3-33)
- “ENT-ASAP-PROF: Enter alarm severity assignment profile” (p. 3-1446)
- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “RTRV-ASAP-PROF: Retrieve ASAP profile” (p. 3-2249)
- “INH-MSG: Inhibit message” (p. 3-2021)
- “REPT ALM: Report alarm” (p. 3-2096)
- “REPT-ALM-ENV: Report alarm environment” (p. 3-2101)

- "REPT EVT: Report event" (p. 3-2121)
- "RTRV-ALM: Retrieve alarm" (p. 3-2231)
- "RTRV-ALM-ENV: Retrieve alarm environment" (p. 3-2238)
- "RTRV-HDR: Retrieve header" (p. 3-2417)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: NE and Transport Surveillance Messages

RTRV-CONN: Retrieve data relevant for network connections

Overview

The **RTRV-CONN** command requests connection information for connections which are addressed via TNAs. This can be connections within one domain or connections via multiple domains requested by TL1 or UNI. This message can be addressed to the ingress node of every involved domain where the specific information of the requested connection are stored in the connection database. The domain ingress node can be the source node of the Network Connection (UNI-N or node receiving the ENT-CONN) or a downstream eNNI node (eNNI-D).

The query must be directed to one of the domain ingress nodes. Querying by the *connID* of an existing ONNS connection of the network connection will return provisioning information about this connection as also signaling information associated with the *connID*.

The command supports querying of all network connections under a *callID* by sending RTRV-CONN with a existing *callID*. When the records of the connection data repository are retrieved by RTRV-CONN with a valid *callID*, each connection record will be in the same format as the RTRV-CONN single *connID* response message format.

Querying by the source TNA will return provisioning information about network connections as also signaling information associated with this TNA as source TNA. This query will be completed only at the originator node (ingress UNI-N or source node receiving this TL1 command) of this network connection.

The command supports querying all connections by sending RTRV-CONN with *connID* = ALL. When all of the records of the connection data repository are retrieved by RTRV-CONN with *connID* = ALL, each connection record will be the same format as the RTRV-CONN single *connID* response message format.

The input parameter settings for *callID*, *connID* and *TNA* from its meaning are mutually exclusive. This exclusion is not explicitly checked.

Minimal one of the input parameters *callID*, *connID* or *TNA* has to be specified in the RTRV request. Otherwise this request is denied. A combination of the input parameters are allowed but this parameters have to match.

REPT CONNCHG is not generated

The **RTRV-CONN** command does not generate a REPT CONNCHG message.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: NA

Abortable: No

Input format

Command syntax: **`RTRV-CONN`**:*srcTid:srcAid:ctag:[callID]:[connID][:specBlock];*

Input parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in RTRV-HDR.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier *srcAid* of the ONNS controller OS command interface port.
   Type: AID.
   Range: snn
   Default: n/a.
3. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″). It should be a unique value for each new command.
4. **`callID`**
   The *callID* indicates a call of a logical call / connection separation (call capability). This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 .. F)
5. **`connID`** – Connection ID
   Connection ID for an existing ONNS connection segment of a UNI or Network connection. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.

Spec Block Parameters

1. **IPv4TNAaddr**
   Identifier of a TNA address in IPv4 format
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0..255
2. **IPv6TNAaddr**
   Identifier of a TNA address in IPv6 format
   Type: Eight hyphen separated hexadecimal numbers
   Range: for each hexadecimal number: 0..FFFF
3. **NSAPTNAaddr**
   Identifier of a TNA address in NSAP format
   Type: Alphanumeric value set (20 byte)
   Range: up to 20 alphanumeric strings with hexadecimal values 00..FF

Output format

Normal system response to RTRV-CONN for connID, Output Syntax

If the RTRV-CONN request completes successfully, then the following normal completion response is returned to the command:

```
   srcTid date time
M  ctag COMPLD
   "[CallID=callID],ConnID=connID,ServiceLevel=servLevel,dir,bandwidth,
      [protType],[remoteTID]:specBlock:
GenLabelRequest=genLabelRequest,Tspec=tspec";
```

System response to RTRV-CONN with connID=ALL for an empty Network Connection Data Repository

If the RTRV-CONN ALL request completes successfully with an empty Network Connection data repository, then the following completion response is returned to the command:

```
   srcTid date time
M  ctag COMPLD
;
```

Output parameters

Command parameters

1. **`srcTid`**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE. See also the definition of *tid* in RTRV-HDR.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`date`**
   Date. The date parameter is set to the date when the output message is generated.
   Type: Value set.
   Range:
   YY-MM-DD, where
   - YY=00-99,
   - MM=01-12,
   - DD=01-31

   Default: current date
3. **`time`**
   Time. The time parameter is set to the time when the output message is generated.
   See also time in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Value set.
   Range:
   HH:MM:SS, where
   - HH=00-23,
   - MM=00-59,
   - SS=00-59

   Default: current time
4. **`ctag`**
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "."). It should be a unique value for each new command.
5. **`completion code`**
   Standard TL1 completion code result.
   Type: Value set.

Range:

- COMPLD: for success

Data continuation is indicated by “>”.

6. **callID**
   callID is an optional parameter for Network Connections setup via UNI.
   The *callID* indicates a call of a logical call / connection separation (call capability).
   This *callID* can be a operator specific identity of Type = 0x01 (source transport network element address is 4 bytes).
   Type: Alphanumeric value set (12 bytes). The first 4 bytes describe the source transport network element address and the following 8 bytes the local identifier.
   Range: string with 24 hexadecimal numbers (0 .. F)
7. **connID – Connection ID**
   Connection ID for an existing ONNS connection segment of a UNI or Network connection. *ConnID* is formed from the *SrcNodeID*, the Ingress Tributary # and an index # where 1675 LambdaUnite MSS uses bay =1= shelf.
8. **servLevel**
   The service level is mapping the connection type of an ONNS connection segment of the Network Connection. The values of the connection type are representing a mapping of the ONNS parameters *connectionClass*, *disjointness* and *grpSetupType* (for Ethernet connections).
   Type: Numeric string.
   Range: 0..255
9. **dir**
   Path Direction Topology:
   - Unidirectional A-Z (1WAY) or
   - Bidirectional (2WAY)

   For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
   Type: Value set.
   Range: 1WAY (default) or 2WAY
10. **bandwidth** – Requested bandwidth / signalRate
    Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
    Type: Value set.
    Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
    Default: None.
11. **protType**
    The *protType* defines the requested protection of advertised abstract links (interdomain as well as intradomain).
    protType is the requested setup parameter (via **ENT-CONN**) and is only visible at the source node if requested.
    Type: Alphanumeric value set.

Range:

- 0x1: only unprotected links shall be considered for routing
- 1+1: only 1+1 MSP protected links shall be considered for routing)
- N/A: all links shall be considered for routing, irr. of their protection type

12. **`remoteTID`**
The *remoteTID* parameter is only visible if provisioned on the eNNI ports. It provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty string (default)
Range: Please refer to the TID definition in the RTRV-HDR description.

13. **`genLabelRequest`**
The general label request defines characteristics of the connection segments between source UNI-C to UNI-N and destination UNI-N to UNI-C or between domains (eNNI-U and eNNI-D).
Type: Alphanumeric value set.
Range:

General Label Request: three dash separated alphanumeric value sets representing

- LSPEncType:
  – PACKET: LSP Encoding Type #1
  – ETHERNET: LSP Encoding Type #2
  – PDH: LSP Encoding Type #3
  – SONET/SDH: LSP Encoding Type #5
  – DW: LSP Encoding Type #6
  – LAMBDA: LSP Encoding Type #8
  – FIBER: LSP Encoding Type #9
  – FCHANNEL: LSP Encoding Type #11
- SwitchingType:
  – PSC-1: Switching Type #1
  – PSC-2: Switching Type #2
  – PSC-3: Switching Type #3
  – PSC-4: Switching Type #4
  – L2SC: Switching Type #51
  – TDM: Switching Type #100
  – LSC: Switching Type #150
  – FSC: Switching Type #200
- G-PID:
  – ETHERNET: G-PID #33

– SONET/SDH: G-PID #34

14. **`tspec`** – Traffic Parameters

    The range of the traffic parameters depends on the used LSP Encoding Type. The *tspec* defines information about the traffic of the connection segments between source UNI-C to UNI-N and destination UNI-N to UNI-C or between domains (eNNI-U and eNNI-D).

    Type: Character string.

    Range for LSP Encoding Type “SONET/SDH”:

    SONET/SDH tspec: five dash separated numeric strings representing SignalType#, RCC#, NCC#, NVC#, MT# with:

    - SignalType #1..12
    - RCC #0,1
    - NCC #0..255
    - NVC #0..63
    - MT #1..64

    Example for an STS-12c: 6-1-4-0-1-0.

    SONET/SDH Tspec parameters “T” and “Profile” is not stored in NVM.

    Range for LSP Encoding Type “ETHERNET”:

    ETHERNET tspec: five dash separated numeric strings representing Token Bucket Rate#, Token Bucket Size#, Peak Data Rate#, Minimum Policed Unit#, Maximum Packet Size# with:

    - Token Bucket Rate #32-bit IEEE floating point number
    - Token Bucket Size #32-bit IEEE floating point number
    - Peak Data Rate #32-bit IEEE floating point number
      (e.g. 0x4CEE6B28 for 1GE)
    - Minimum Policed Unit #1..FFFFFFFF
    - Maximum Packet Size #1..FFFFFFFF

Spec Block Parameters

1. **`destLabel`**

   The destination label defines together with the destination logical port ID the associated egress *connPoint* that connects the destination node of the Network Connection. The *destLabel* is defined in KLM notation. *destLabel* is the requested setup parameter (via ENT-CONN or UNI) and is only visible at the source node of the Network Connection.

   Type: Character string.

   Range:

   Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:

   - S #0..65535
   - U #0..15

- K #0..15
- L #0..15
- M #0..15

Example for an STS-12c which uses the 9th STS-3 as first timeslot: 9-0-0-0-0.
Destination Label 65535-15-15-15-15 is reserved for GbE connections.

2. **`destLogicalPortID`**
   destLogicalPortID is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
3. **`destIPv4TNAAddr`**
   destIPv4TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is *IPV4TNAADDR*.
4. **`destIPv6TNAAddr`**
   destIPv6TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is *IPV6TNAADDR*.
5. **`destNSAPTNAAddr`**
   destNSAPTNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *destTNAType* parameter value (via ENT-CONN or UNI) is *NSAPTNAADDR*.
6. **`routingList`**
   The *routingList* describes the level 1 routing described as a list of level 1 nodeIDs and linkIDs The *routingList* retrieved from an intermediate or destination domain will not show the *localAdvNodeID – localLinkID* combination of preceding domains.
   Type: Character string.
   Range:

   Hyphen separated list of numeric string pairs, where the first numeric string represents a *localAdvNodeID* and the second a *localLinkID* with:

   - *localAdvNodeID*: four dot separated numeric string, each numeric string with the values 0..255
   - *localLinkID*: numeric string, with the values 0..42946967295.

   at maximum 10 *localAdvNodeIDs* and 10 *localLinkIDs*.
   Example for a *routingList* describing level 1 Routing via 3 domains at the source domain: 126.43.231.4-643-1.232.3.1-4209-1.4.23.8-5.
7. **`exclLinkResCls`**
   The exclude list of *linkResCls* is used to exclude interdomain links from the interdomain routing calculation. *exclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
   Type: Alphanumeric value set (4 bytes), every bit defining one link class
   Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF

8. **`inclLinkResCls`**
   The include list of *linkResCls* is used to define interdomain links which has to be included for interdomain routing calculation. *inclLinkResCls* is the requested setup parameter (via ENT-CONN) and is only visible at the source node of the Network Connection.
   Type: Alphanumeric value set (4 bytes), every bit defining one link class
   Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF

9. **`ingressLSP`**
   Local Connection ID for the connection segment of a Network Connection between source UNI-C to UNI-N or eNNI-U to eNNI-D.
   srcLSP is formed from the Tunnel_ID of the UNI_IPv4_SESSION/eNNI_IPv4_SESSION object and the LSP_ID of the LSP_TUNNEL_IPv4_SENDER_TEMPLATE of this connection segment.
   Type: character string of length up to 1
   Range:

   TunnelID-LSPID, with

   - TunnelID: numeric string: 0..65535
   - LSPID: numeric string: 0..65535

   Example: 2319-321

10. **`policyData`**
    Policy data may in general include credentials identifying users or user classes, account numbers, limits, quotas, etc. to support access policies.
    Type: Alphanumeric value set (60 bytes)
    Range: up to 60 alphanumeric characters with hexadecimal values 00..FF

11. **`srcLabel`**
    The source label defines together with the source logical port ID the associated ingress *connPoint* that connects the source node of the Network Connection. The *srcLabel* is defined in KLM notation.
    srcLabel is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
    Type: Character string.
    Range:

    Generalized Label: five dash separated numeric strings representing S#,U#,K#,L#,M# with:

    - S #0..65535
    - U #0..15
    - K #0..15
    - L #0..15
    - M #0..15

    Example for an STS-12c which uses the 9th STS-3 as first timeslot: 9-0-0-0-0.
    Source Label 65535-15-15-15-15 is reserved for GbE connections.

12. **`srcLogicalPortID`**
    Logical Port Identifier related to the TNA address. *srcLogicalPortID* is the requested setup parameter (via ENT-CONN or UNI) and is only visible on the source node of the Network Connection.
    Type: numeric value set
    Range: 0 to 4294967295
13. **`srcIPv4TNAAddr`**
    srcIPv4TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *srcTNAType* parameter value (via ENT-CONN or UNI) is *IPV4TNAADDR*.
14. **`srcIPv6TNAAddr`**
    srcIPv6TNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *srcTNAType* parameter value (via ENT-CONN or UNI) is *IPV6TNAADDR*.
15. **`srcNSAPTNAAddr`**
    srcNSAPTNAAddr is an optional parameter and is only visible on the source node of the Network Connection if the requested *srcTNAType* parameter value (via ENT-CONN or UNI) is *NSAPTNAADDR*.
16. **`tunnelIfID`**
    All SDH/SONET connections belonging to the same VCAT group are identified by the LSP Tunnel Interface ID.
    Type: character string of length up to 24
    Range:
    LSRRouterID&InterfaceID - with:
    - *LSRRouterID*: four dot separated numeric strings: each string 0..255
    - *InterfaceID*: Numeric string 0..4294967295

    Example: 117.32.234.13&32439

Example input/output

Input/Output example for a Network Connection

The following example shows the successful completion of a RTRV-CONN command by the Network Element for an unprotected STS-12 Network Connection at the source domain:

```
RTRV-CONN:LT-ONNS9:snn:113456::135.13.196.203&1-6-5-3-1&1;

   LT-ONNS9 2006-01-01 08:00:00
M  113456 COMPLD
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.196.203&1-6-5-3-1&1,
ServiceLevel=0,2WAY,STS12c,0x1,LT-ONNS22:IngressLSP=2319-203&
SRCIPV4TNAADDR=135.13.3.119&SrcLogicalPortID=1019&SrcLabel=9-0-0-0-0&
DESTIPV4TNAADDR=135.225.43.128&DestLogicalPortID=2&DestLabel=4-0-0-0-0&
RoutingList=126.43.231.4-13.121.39.21-1.232.3.1-12.38.217.4-1.4.23.8:
GenLabelRequest=5-100-34,Tspec=6-1-4-0-1-0″;
```

Input/Output example for a Network Connection

The following example shows the successful completion of a RTRV-CONN command by the Network Element for an unprotected STS-12 Network Connection at the intermediate domain:

```
RTRV-CONN:LT-ONNS22:snn:113454::135.13.122.20&1-6-4-3-1&1;

   LT-ONNS22 2006-01-01 08:00:00
M  113454 COMPLD
   ″CallID=A123B3EF1345A123B3EF1345,ConnID=135.13.122.20&1-6-4-3-1&1,
ServiceLevel=0,2WAY,STS12c,0x1,LT-ONNS45:IngressLSP=2125-2122&
RoutingList=1.232.3.1-12.38.217.4-1.4.23.8:GenLabelRequest=5-100-34,
Tspec=6-1-4-0-1-0″;
```

Input/Output with multiple Network Connections

The following structure shows the successful completion of a RTRV-CONN command with multiple Network Connections specified by a CallID:

```
RTRV-CONN:LT-ONNS22:snn:113454:A123B3EF1345A123B3EF1345:;

   LT-ONNS22 2006-01-01 08:00:00
M  113454 COMPLD
   ″CallID=A123B3EF1345A123B3EF1345,
    .
    .
    Tspec=6-1-4-0-1-0″;
    ...
    ″CallID=A123B3EF1345A123B3EF1345,
    .
    .
    Tspec=6-1-4-0-1-0″;
    ...
    ″CallID=A123B3EF1345A123B3EF1345,
    .
    .
    Tspec=6-1-4-0-1-0″;
```

Error responses

General error responses

A command request without input parameters is denied with:

Error code: `IPMS – Input, Parameter MiSsing, <parametername> is missing`

Validation rule: Input Parameter *connID* not valid

If the value for the input parameter *connID* is not valid, the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, invalid connection ID`

Validation rule: Input Parameter *TNA address* not valid

If the value for the input parameter *TNA address* is not valid, the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, invalid TNA address`

Validation rule: Input Parameters not matching

If the value for the input parameters *connID*, *TNA address* or *callID* are not matching, the following error response is returned.

Error code:
`IDNV – Input, Data Not Valid, input parameters are not matching`

Validation rule: Input Parameter *callID* not valid

If the value for the input parameter *callID* is not valid, the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, invalid callID`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
- "DLT-CONN: Delete network connection" (p. 3-139)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT-PTHCHG notification" (p. 3-2149)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT CONNCHG notification" (p. 3-2104)

RTRV-CRS: Retrieve cross-connection

The **RTRV-CRS**-*modifier* command can be initiated by a user to retrieve information on existing cross-connections in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-CRS**-*modifier*:*tid*:[*in_aid*]:*ctag*[::*ppgname*];

Either the parameter *in_aid* or *ppgname* must be present – refer to the ″Detailed behavior description″ section for details.

Input parameters

Important! Do not use *ppgname* in **RTRV-CRS** requests. Just use the *in_aid* parameter in **RTRV-CRS** requests.

Command parameters

1. **modifier**
This parameter indicates the cross-connection rate on which the retrieve cross-connection command acts.
A distinction of the *modifier* usage has to be made between the retrieval (RTRV-CRS-*modifier*) and the other CRS-related commands (ENT-CRS-*modifier*, DLT-CRS-*modifier*, ED-CRS-*modifier*, ENT-ROLL-*modifier*):
   - The establishment (ENT-CRS-*modifier*), removal (DLT-CRS-*modifier*), modification (ED-CRS-*modifier*) or rolling (ENT-ROLL-mod) of cross-connections is independent of the provisioned port interface-standard. I.e., either the SDH-rates (VC3, VC4 etc.) or the corresponding SONET-rates (STS1, STS3 etc.) can be used as modifier and no command validation between port interface-standard and *modifier* will be performed.
   - For the RTRV-CRS-*modifier* command the following holds:
     Again, no command validation between port interface-standard and *modifier* will be performed. The output parameter *rate*, however, is filled with the rate which was used with the RTRV-CRS-*modifier* command.
     Example: if a cross-connection was entered as ENT-CRS-VC3 then a RTRV-CRS-ALL as well as a RTRV-CRS-VC3 as well as RTRV-CRS-STS1 will return this cross-connection with the *rate* parameter VC3.

   Rate restriction for AU3/AU4 Conversion Cross-Connections:
   AU3/AU4 conversion cross-connections are cross-connections between HO an LO tributaries. From cross-connection processing point of view, these are tributaries

with different rates, although the tributaries provide equal bandwidth. Two different modifiers are defined for conversion cross-connections, one for HO input tributary to LO output tributary and another for LO input tributary to HO output tributary. The input and output tributary rates must fit to the modifier.
Type: Alphanumeric value set.
Range:

- ALL
- SDH *modifier*: LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C.
- SONET *modifier*: LOVT1, STS1, STS3, STS12, STS48, STS192.

Conversion Cross-Connections:

- AU3TU3 (HO VC3/STS1 input to LO VC3 output)
- TU3AU3 (LO VC3 input to HO VC3/STS1 output)

Note that the modifier value ALL can only be used to retrieve all high-order or to retrieve all low-order and conversion cross-connections at once, due to different wildcard AIDs. The **RTRV-CRS-ALL** command considers conversion cross-connections with LO AIDs.

2. **tid** – Target Identifier
Target Identifier. Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.

3. **in_aid** – Input Access Identifier
This parameter specifies the source of the cross-connection.
Each cross-connection can be addressed by its input or output tributary. Each atomic cross-connection associated with that input or output will be returned except for input tributaries being part of a path-protected cross-connection (here only one leg will be reported).
The specified input value of this parameter may be the in_aid or the out_aid parameter of the response depending on how the cross-connection had been created:

- for 1WAY/1WAYPROT connections, the replied in_aid is always a source of the cross-connection,
- for 2WAY cross-connections, the replied in_aid is the same as it has been specified as *in_aid* within the **RTRV-CRS** command. If wildcards are used in the AID, then either of the input/output AIDs of the cross-connections will be returned as *in_aid* in the response.

Examples:
ENT-CRS-VC4:NEname:1-1-#-#-1-1-1,1-1-#-#-2-1-1:123456:2WAY;
RTRV-CRS-VC4:NEname:1-1-#-#-2-1-1:123457;
M 123457 COMPLD ″1-1-#-#-2-1-1,1-1-#-#-1-1-1:VC4.....
;
ENT-CRS-VC4:NEname:1-1-#-#-3-1-1,1-1-#-#-4-1-1:123458:1WAY;
RTRV-CRS-VC4:NEname:1-1-#-#-4-1-1:123459;
M 123459 COMPLD ″1-1-#-#-3-1-1,1-1-#-#-4-1-1:VC4.....
;

In case of lower-order cross-connections, the source of the cross-connection has to be provided logically associated a transmission port pack although the physical cross-connection is only present LOXC pack internally. The NE is in charge to map physical to logical representation.
Example:
ENT-CRS-LOVC3:NEname:1-1-#-#-1-1-1-1,1-1-#-#-39-16-1-43:123456:2WAY; M 123456 COMPLD;
RTRV-CRS-LOVC3:NEname:1-1-#-#-1-1-1-1:123457; M 123457 COMPLD ″1-1-#-#-1-1-1-1,1-1-#-#-39-16-1-43:LOVC3;

In the case of conversion cross-connections, the *in_aid* specified in the **RTRV-CRS** command must fit to the modifier:

- *modifier* = AU3TU3:*in_aid* = HO tributary AID
- *modifier* = TU3AU3:*in_aid* = LO tributary AID

Conversion cross-connections can be addressed by their input or output tributaries. Each conversion cross-connection associated with that input or output will be returned. The specified input value of *in_aid* in RTRV-CRS may be the *in_aid* or the *out_aid* parameter of the response depending on how the cross-connection had been created:

- for 1WAY conversion cross-connections, the replied *in_aid* is always a source of the cross-connection,
- for 2WAY cross-connections, the replied in_aid is the same as it has been specified as *in_aid* within the **RTRV-CRS** command. If wildcards are used in the AID, then either of the input/output AIDs of the cross-connections will be returned as *in_aid* in the response. The returned *in_aid* is independent, whether the wildcard AID in RTRV-CRS specifies a HO or LO tributary.

Conversion Cross-Connection Examples:
ENT-CRS-AU3TU3:NEname:1-1-#-#-1-1-1,1-1-#-#-2-1-1-1:123456:2WAY;
RTRV-CRS-AU3TU3:NEname:1-1-#-#-1-1-1:123457;
M 123457 COMPLD ″1-1-#-#-1-1-1,1-1-#-#-2-1-1:AU3TU3…2WAY…
;
RTRV-CRS-TU3AU3:NEname:1-1-#-#-2-1-1-1:123457;
M 123457 COMPLD ″1-1-#-#-2-1-1-1,1-1-#-#-1-1-1:TU3AU3…2WAY…
;
RTRV-CRS-TU3AU3:NEname:1-1-#-#-all-all-all-all:123458;
M 123457 COMPLD ″1-1-#-#-1-1-1,1-1-#-#-2-1-1-1:AU3TU3…2WAY…
;
ENT-CRS-TU3AU3:NEname:1-1-#-#-3-1-1-1,1-1-#-#-4-1-1:123458:1WAY;
RTRV-CRS-TU3AU3:NEname:1-1-#-#-3-1-1-1:123459;
M 123459 COMPLD ″1-1-#-#-3-1-1-1,1-1-#-#-4-1-1:TU3AU3…1WAY…
;
RTRV-CRS-AU3TU3:NEname:1-1-#-#-4-1-1:123459;
M 123459 COMPLD ″1-1-#-#-3-1-1-1,1-1-#-#-4-1-1:TU3AU3…1WAY…
;
Type: GenTribAID or GenLOTribAID.

Range: The *all*-parameter is accepted for the slot-, port- and higher-order and (if applicable) lower-order trib-part of the AID. The *usall*-parameter is accepted additionally for the slot-part of the AID.
The *vall*-parameter is accepted for the port part of the AID alternatively to the *all*-parameter as long as a virtual concatenated tributary will be addressed. Entities having the *all*-parameter as port-part of the AID reside e.g. on OP***n***, EP***n***, TransMUX. Entities having the *vall*-parameter as port-part of the AID reside e.g. on GE***n***, OPT2G5, ES64U.

For example the following *in_aids* are valid:

- 1-1-#-#-all-all-all specifying all high-order tribs of all ports in the whole shelf
- 1-1-#-#-2-1-all specifying all tribs of the first port of the pack in slot 2
- 1-1-#-#-3-vall-all specifying all tribs of all port of the (GE***n***/ES64U) pack in slot 3
- 1-1-#-#-all-all-all-all specifying all low-order of all high-order tribs for all ports in the shelf

However, the following example shows an invalid *aid*:
1-1-#-#-all-vall-all-all there is no low-order tributary on e.g. GE***n***, OPT2G5, ES64U packs.

4. **`ctag`** – Correlation Tag
See definition of *ctag* in “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

In response to a valid RTRV-CRS-mod command, the following output is returned:

```
   sid date time
M  ctag COMPLD
   ″in_aid,out_aid:rate[,,in_aid2]:spec_block″
       ...
       ...
   ″in_aid,out_aid:rate[,,in_aid2]:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`in_aid`** – Input Access Identifier

Description: The parameter *in_aid* specifies the tributary which is the source of the cross-connection. In detail:

- In case of a 1-way cross-connection this parameter specifies the source of the cross-connection.
- In case of a 2-way cross-connection this parameter is identical to the one used as input parameter (please refer also to the description of the input parameter *in_aid* of the "RTRV-CRS: Retrieve cross-connection" (p. 3-2306) command).
- In case that a wildcard AID has been used to retrieve 2-way cross-connections, then either of the input/output AIDs of the cross-connections will be returned as *in_aid* in the response. For conversion cross-connections the returned *in_aid* is independent, whether the wildcard AID in RTRV-CRS specifies a HO or LO tributary.
- In case of a path-protected cross-connection this parameter may specify either the source of the working leg or the source of the protection leg – the distinction is made via the spec-block parameter *repleg* (see below). If *in_aid* specifies the working-leg part than *in_aid2* (see below) specifies the protection-leg part and vice versa.

Type: GenTribAID or GenLOTribAID.
STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs;
VCGNumAID are used for high-order tributaries on e.g. GE***n***, OPT2G5 packs;
GenLOTribAID are used for low-order tributaries on I/O packs e.g. OP***n***, EP155.
Range: n.a. (*all* or *usall* or *vall* values shall never occur as part of an output parameter).

2. **`out_aid`** – Output Identifier
   This parameter specifies the tributary which is the sink of the cross-connection.
   Type: GenTribAID or GenLOTribAID
   STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs;
   VCGNumAID are used for high-order tributaries on e.g. GE***n***, OPT2G5 packs;
   GenLOTribAID are used for low-order tributaries on I/O packs e.g. OP***n***, EP155.
   Range: n.a. (*all* or *usall* or *vall* values shall never occur as part of an output parameter).

3. **`in_aid2`** – Input Access Identifier 2
   This parameter is applicable to and is present for path-protected cross-connections only and specifies either the source of the protection leg or the source of the working leg.

This depends on the value of the *repleg* parameter:

- E.g. for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSW the worker leg is given by IN1 -> OUT and the protection leg is given by IN2 -> OUT;
- for the combination: *in_aid* = IN1, *out_aid* = OUT, *in_aid2* = IN2 and *repleg*=1WAYPSP the worker leg is given by IN2 -> OUT and the protection leg is given by IN1 -> OUT.

Thus *in_aid* and *out_aid* specify one leg of the legpair of a path-protected cross-connection, while *in_aid2* and *out_aid* specify the second leg.
Type: STS1NumAID, TMuxAPTNumAID, or GenLOTribAID.
STS1NumAID are used for high-order tributaries on e.g. OP***n***, EP***n*** packs; TMuxAPTNumAID are used for asynchronous high-order tributaries on TransMUX packs, GenLOTribAID are used for low-order tributaries on e.g. OP***n***, EP155.
Note: Using VCGNumAID (when using e.g. GE***n*** packs) is NOT applicable as those packs can not be the input to a path-protected cross-connection. Therefore specific AID types are mentioned instead of GenTribAID.
Range: n.a. (*all* and *usall* values shall never occur as part of an output parameter).

4. `rate`
This parameter indicates the cross-connection rate. For a detailed description please refer to the input parameter *modifier*.
Type: Alphanumeric value set.
Range:

- LOVC12
- LOVC3
- VC3
- VC4
- VC44C
- VC416C
- VC464C
- LOVT1
- STS1
- STS3
- STS12
- STS48
- STS192.

Conversion Cross-Connections:

- AU3TU3 (HO VC3/STS1 input to LO VC3 output)
- TU3AU3 (LO VC3 input to HO VC3/STS1 output)

Spec block parameters

1. **loca** – Location-a

   This parameter specifies the TID of the node at which service is added to the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and DROP connections only. The *loca* parameter value is an empty string for ADD and non-BLSR connections.

   A 2-way cross-connection is considered to be a DROP if the *out_aid* is a non BLSR tributary.

   The figure in "2-way scenario" (p. 3-2322) depicts the 2-way scenario (the 3 NEs should be seen as part of an BLSR).

   Inter Ring Cross-connections with collapsed node: Note that beside the above described THROUGH case there can be also another situation where both *loca* and *locz* are different from the empty string: If two rings are interconnected in a collapsed node then *loca* refers to the (add-node of the) ring where the signal is dropped from, and *locz* refers to the (drop-node of the) ring where the signal is added to. In such cases the cross-connection cannot be provisioned as bidirectional, thus the information is still unambiguous.

   A note to the loca2 parameter: Although *loca2* (Location-a-2) is a parameter which is used in the **ENT-CRS** command, *loca2* is not retrieved explicitly. This is due to the fact that for path-protected cross-connections – and only for those this *loca2* is applicable – the retrieval will be on a leg-basis and not on a leg-pair basis. In other words: If a path-protected cross-connection is retrieved two entries will be sent, one for the working leg (incl. the *loca* and *locz* parameter) and one for the protection leg (incl. the *loca* parameter and the same *locz* parameter, and in this case the *loca* returned is the *loca2* as entered when establishing the path-protected cross-connection).

   loca is also required for EXTRA TRAFFIC connections. In contrast to incoming references from other commands to this whole chapter it is exclusively applicable for the command **RTRV-CRS**. For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of *loca* refer to the parameter *sqfa*.

   For lower-order cross-connections the parameter *loca* is the TID of that ring node in the direction towards the specified input tributary (upstream), where the HO path termination source function resides.

   If there is no such function in an upstream ring node then *loca* is the TID of the node where the related HO circuit is first added to the ring. The figure "Lower-order cross-connections" (p. 3-2322) shows an example of a LO path through as part of a ring.

   The LO path is added to/dropped from the ring in NE-1 and NE-2. 2-way cross-connections are considered to be either an Add or a Drop connection depending whether the input or the output tributary is at the ring port. In NE-3 there is a HO through connection, which contains the LO path. In NE-3 there is no HO path termination. In NE-4 there is a LO through connection. Consequently, there are HO path terminations in NE-4.

The fact that *loca* refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same *loca* value.
- Any LO through-connection will be treated as add/drop. loca will identify the node of the through connection, and not the node where the LO service is added to the ring.

At the lower-order side of a conversion cross-connection, the same rules for *loca* is applicable as for lower-order cross-connections.
Type: TID or empty string.
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string ″″:
Note: In output direction the empty string appears as \″\″.

2. **`locastate`** – Squelch Map Data Status for Location a
   The parameter *locastate* indicates the validity and the source of the *loca* value.
   Type: Alphanumeric value set.
   Range:
   - AUTO-VALID
   - AUTO-INVALID
   - USER-OVERRIDE
   - MANUAL-VALID
   - MANUAL-INVALID
   - UNKNOWN (default).
3. **`locz`** – Location-z
   This parameter specifies the TID of the node at which service is dropped from the BLSR. This parameter is required for BLSR THROUGH, INTER-BLSR and ADD connections. The *locz* parameter value is an empty string for DROP and non-BLSR connections.
   A 2-way cross-connection is considered to be a ADD if the *in_aid* is a non BLSR tributary.
   Refer to the figure in “2-way scenario” (p. 3-2322) for further explanation.
   locz is also required for EXTRA TRAFFIC connections. For extra traffic the value will be generated automatically and thus it is only retrievable. For information about the automatic generation of *locz* refer to the parameter *sqfz*.
   For lower-order cross-connections the parameter *locz* is the TID of that ring node in the direction towards the specified output tributary (downstream), where the HO path termination sink function resides.
   If there is no such function in a downstream ring node then *locz* is the TID of the node, where the related HO circuit is last dropped from the ring. Refer to the figure in “Lower-order cross-connections” (p. 3-2322) for further explanation.

The fact that locz refers to the HO path terminations has the following consequences:

- All LO cross-connections through the same HO path termination at the MS-SPRing port have the same *locz* value.
- Any LO through-connection will be treated as add/drop. *locz* will identify the node of the through connection, and not the node where the LO service is dropped from the ring.

At the lower-order side of a conversion cross-connection, the same rules for *locz* is applicable as for lower-order cross-connections.
Type: TID or empty string.
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string ″″:
Note: In output direction the empty string appears as \″\″.

4. **`loczstate`** – Squelch Map Data Status for Location z
   This parameter indicates the validity and the source of the *locz* value.
   Type: Alphanumeric value set.
   Range:
   - AUTO-VALID
   - AUTO-INVALID
   - USER-OVERRIDE
   - MANUAL-VALID
   - MANUAL-INVALID
   - UNKNOWN (default).
5. **`lpbkstat`** – Loopback-Status
   This parameter specifies whether a cross-connection loopback exists on the input tributary or the output tributary of the cross-connection.
   Type: Alphanumeric value set.
   Range: YES, NO (default).
6. **`omode`** – Output Mode
   This parameter specifies the output mode of the cross-connection:
   - for a 1-way cross-connection the output is clearly defined by the destination (*out_aid*)
   - for a 2-way cross-connection the output is defined as the source-to-destination direction (*in_aid* to *out_aid*)
   - NORM: this is the normal output i.e. the signal from the *in_aid* is passed through to the *out_aid*;

- IDLE/UNEQ: this mode outputs an unequipped signal;
- AIS: this mode outputs an AIS signal.

omode and adaptive-mode: In case a SONET-port is provisioned to adaptive-mode setting the *omode* to IDLE/UNEQ or AIS will be lead to transmitting the corresponding signal (UNEQ or AIS) on STS1 base for all the STS1 tributary comprising the STS-N signal. Thus for an output of an STS3 cross-connection, if either AIS or IDLE/UNEQ is selected each of the 3 STS1 will output the AIS or IDLE/UNEQ signal. When switching the *omode* back to NORM the original STS3 cross-connection-signal will re-appear again.
omode in case of path-protected cross-connection: In case of a path-protected cross-connection the *omode* affects the output of the path selector and is thus not setable on a per leg base. In other words the *omode* insertion is done after (downstream direction) the selector function.
Type: Alphanumeric value set.

Range:

- NORM (default)
- IDLE/UNEQ
- AIS

7. **`onnsConnControl`** – ONNS Connection Control
This parameter indicates for cross-connections of *xconcls* NN, whether the ONNS connection a cross-connection is assigned to is controlled by ONNS directly or by an external control (setup via UNI or eNNI or as SPC connection).
For cross-connections of *xconcls* TRADITIONAL the value UNKNOWN is reported for this parameter.
The value UNKNOWN shall also be reported for cross-connections of *xconcls* NN, the onnsConnControl of which cannot be determined.
Type: Alphanumeric value set.

Range:

- UNKNOWN
- ONNS
- EXT

8. **`onnsPathID`** – ONNS Path Identity
This parameter indicates for cross-connections of *xconcls* NN the path ID of the ONNS path a cross-connection is assigned to.
For cross-connections of *xconcls* TRADITIONAL an empty string is reported for this parameter.
Type: character string of length up to 20 or empty string
Range:

nodeID-Index

- nodeID: four dot separated numeric strings: each string 0..255
- Index: Numeric string 0-65535

Example: 29.13.196.203-65111
Default: empty string

9. **`repleg`** – Reported Leg Pair

This parameter provides compact information about the leg(s) of cross-connections and is derived, as shown in the table below, from a combination of the following elementary parameter:

- atomic cross-connection topology
- leg status
- path-protected group working leg (if applicable i.e. being part of a protection group).

Atomic cross-connection Topology	Path Protection Leg Role	Leg Status	Reported Legpair (*repleg*)
point-to-point	n.a.	1-way	1WAY
point-to-point	n.a.	2-way	2WAY
path-protected	W	1-way	1WAYPSW
path-protected	P	1-way	1WAYPSP

Atomic cross-connection topology:
Two atomic cross-connection topologies are supported:

- Point to point: The point to point cross-connection consists of one leg only, which is defined by its source-trib (*in_aid*) and its destination-trib (*out_aid*) and its particular transmission rate.
- path-protected: The path-protected cross-connection consists of a path protection group with two legs, a working leg and a protection leg – the declaration of working respectively protection is made when establishing the cross-connection (ENT-CRS command).

  The two legs are defined by two input tributaries and one output tributary – thus one input tributary and the output tributary form the working leg while the other input tributary and the output tributary form the protection leg.

  Note that:

  both input tributaries must belong to ports with the same interface standard (either SONET or SDH), due to different switch criteria;

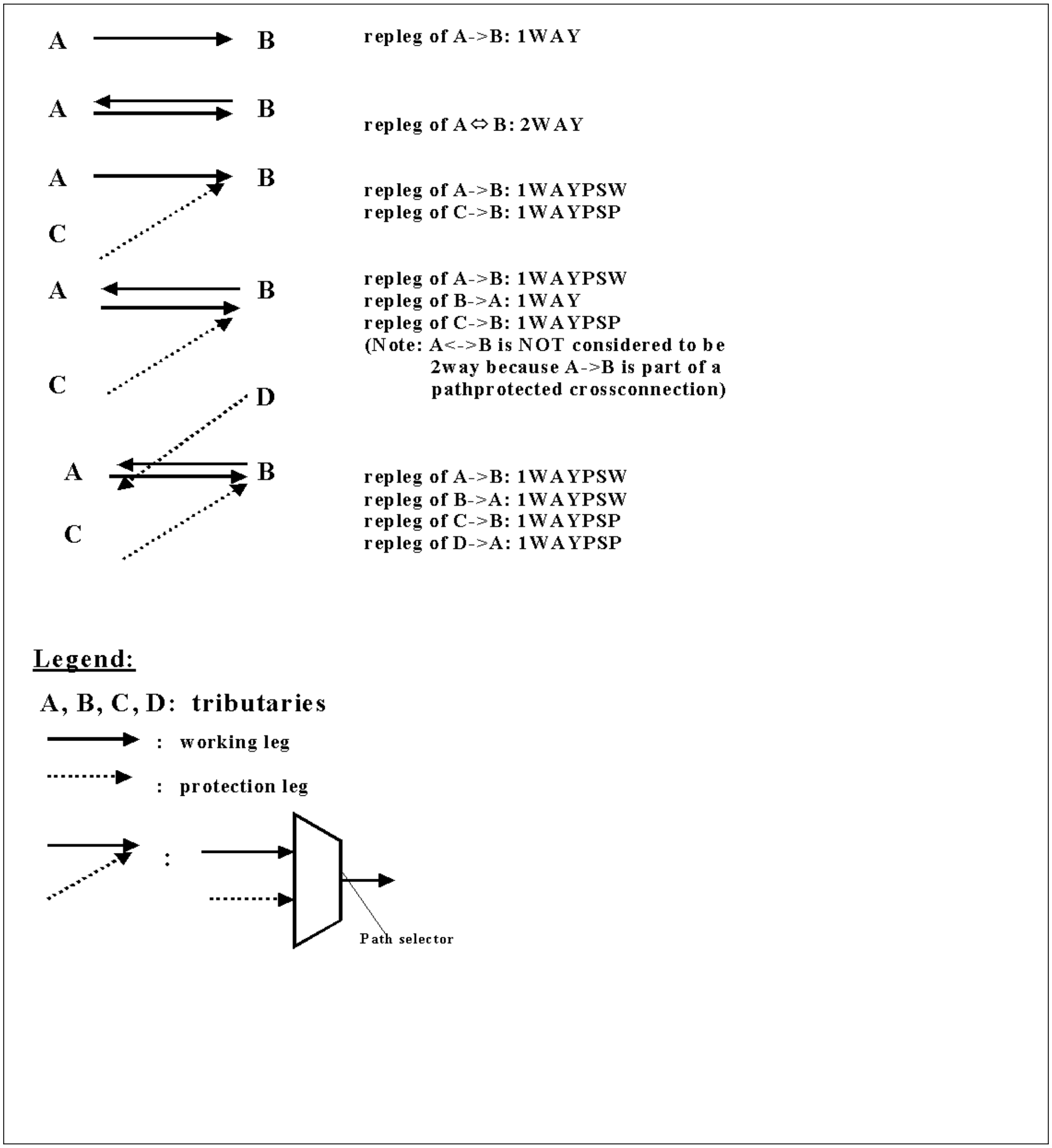
  both legs must be at the same cross-connection rate.

Leg status:
The leg status indicates whether a leg is monitored and reported as a 1-way cross-connection or a leg pair i.e. a 2-way cross-connection.
1-way: The leg status for a path-protected cross-connection is always defined as 1-way.

2-way: The leg status for a (non path-protected) cross-connection is set to 2-way if all the following conditions are fulfilled: there are 2 cross-connections in opposite direction between the same tributaries; both of these cross-connections have the same rate (independent if the SONET or SDH rate was used when establishing i.e. VC3 is equivalent to STS1, VC4 is equivalent to STS3 etc); the source and destination TIDs do match i.e. the source TID of one leg is the destination TID of the other leg and vice versa (this condition of course holds only in case of BLSR cross-connections).
Note that the criteria for a 2-way leg-pair do not depend on whether the two legs were set up at the same time, nor on whether either leg was set up as part of a 2-way cross-connection topology.

Path Protection leg role:

This indicates if the leg of a path protection group is defined as the working (W) leg or the protection (P) leg.
1WAYPSW and 1WAYPSP stands for 1-way path switched working and 1-way path switched protection respectively.
The following figure shows examples for the *repleg* parameter for several cross-connection configurations including compound cross-connection.



Range

- 1WAY
- 2WAY
- 1WAYPSW
- 1WAYPSP.

The values 1WAYPAP and 1WAYPAW are not supported in the initial release.

10. **`rtnomode`** – Return Output Mode

This parameter specifies the output mode for the return direction of the cross-connection.

- for a 2-way cross-connection the return output mode is defined as the destination-to-source direction (*out_aid* to *in_aid*) For the meaning of the values please refer to *omode* description.
- for a 1-way cross-connection and path protected cross-connection this parameter is omitted.

Type: Alphanumeric value set.
Range: NORM (default), IDLE/UNEQ, AIS.

11. **`rtntaccstat`** – Return Test Access Status
This parameter indicates indicates whether the input tributary of the return direction (destination-to-source) of a 2-way cross-connection is cross-connected or bridged to a test access tributary of a test session. In the case of a 1-way cross-connection the parameter is omitted.
Type: Alphanumeric value set.
Range: YES, NO (default).
12. **`rtnxcappl`** – Return cross-connection application
This parameter is a numeric value which indicates the application for which the return cross-connection is used in the context of compound cross-connection topologies.
   - for a 1-way cross-connection this parameter is not applicable and will thus be omitted.
   - for a 2-way cross-connection the parameter is present.

   Type: same as *xcappl*.
Range: same as *xcappl*.
Default: The default value is 0.
13. **`rtnxcnum`** – Return cross-connection number
This parameter is a multiple-digit number identifying the cross-connection number for the return direction (destination to source i.e. *out_aid* to *in_aid*) of a 2-way cross-connection.
   - for a 1-way cross-connection this parameter is not applicable and will thus be omitted.
   - for a 2-way cross-connection the parameter is present.

   Type: same as *xcnum*.
Range: same as *xcnum*.
Default: same as *xcnum*
14. **`taccstat`** – Test Access Status
This parameter indicates whether the input tributary of a 1-way cross-connection or the input tributary of the source-to-destination direction of a 2-way cross-connection is crossconnected or bridged to a test access tributary of a test session.
Type: Alphanumeric value set.
Range: YES, NO (default).

15. **`xcappl`** – cross-connection application
    This parameter is a numeric value which indicates the application for which the cross-connection is used in the context of compound cross-connection topologies.
    The parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection.
    The numeric values are assigned to appropriate text strings (see Range below) and the manager should display these value in case of a "RTRV-CRS: Retrieve cross-connection" (p. 3-2306).

    As a possible assignment proposal the following may serve:

    - 0) Unknown
    - 1) 1-Way Point-to-Point
    - 2) 1-Way Path-Protected
    - 3) 1-Way Adjunct Path-Protected
    - 4) 1-Way Conversion Cross-Connection
    - 10) 2-Way Point-to-Point
    - 11) 2-Way Conversion Cross-Connection
    - 20) UPSR (or SNCP Ring) Add, Drop
    - 21) UPSR (or SNCP) Ring-to-Ring, Single Node Interconnection, Same NE
    - 22) UPSR (or SNCP) Drop, 1-Way Broadcast
    - 23) UPSR (or SNCP Ring) Add, Drop connection (Broadcast)
    - 30) Logical Ring (or SNCP) Add, Drop
    - 40) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Node
    - 41) Ring Interworking, Drop-and-Continue, BLSR (or MS-SPRing) Primary Nodes in same NE
    - 42) Ring Interworking, Dual Transmit, BLSR (or MS-SPRing) Terminating Node
    - 50) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring)
    - 51) Ring Interworking, Drop-and-Continue, UPSR (or SNCP Ring) Nodes in same NE
    - 60) 1-Way Broadcast -- note: this is with N greater than or equal to 2.

    Type: numerical value set.
    Range: {0, 1, 2,…, 255}.
    Default: The default value is 0.

16. **`xcnum`** – cross-connection number
    This parameter is a multiple digit number, identifying each leg in a specific compound cross-connection. (Compound cross-connection are a composition of several atomic cross-connection in order to form complex scenarios like e.g. described in ″Example input/output″ section of the **`RTRV-CRS`** command like UPSR, DRI etc.).

Like the *xcappl* parameter the *xcnum* parameter is not further processed in the NE, but only stored (when establishing or modifying the cross-connection) and returned when retrieving the cross-connection. It may be used by a manager to characterize and group cross-connections belonging to complex cross-connection topology (e.g. compound cross-connections as used in ring interworking scenarios).

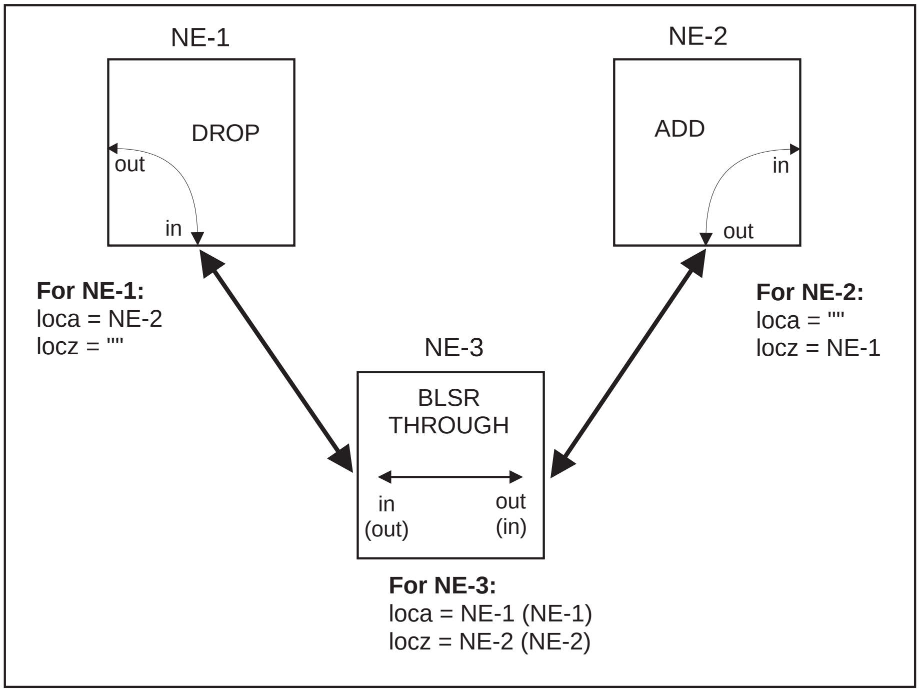
The following format for the twelve digits is recommended:

- two digits for the bay
- one digit for the shelf
- two digits for the slot
- two digits for the port
- three digits for the high-order tributary
- two digits for the low-order tributary.

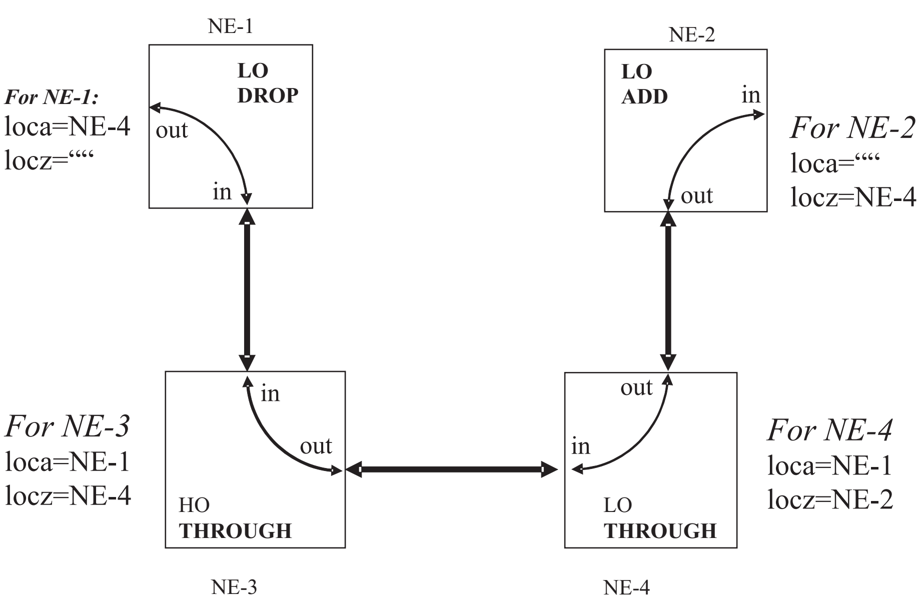
If a high-order tributary is addressed then the last two digits are set to 00. The digits values pertain to the *out_aid*.
Type: numeric value set (12 digit).
Range: 000000000000 .. 999999999999.
Default: The default value is 000000000000.

17. **`xconcls`** – cross-connection class
This parameter identifies, whether the cross-connection has been setup in the TRADITIONAL way by an operator via the **`ENT-CRS`** command or via the NN application as part of an ENT-PATH command. This parameter therefore also indicates who is the owner of the cross-connection and can modify or delete it.
Note: Refer to the NN TL1 catalogue for the **`ENT-PATH`** command.
Type: Alphanumeric value set.
Range: TRADITIONAL, NN.
For low order cross-connections, the value range is limited to TRADITIONAL.

2-way scenario



Lower-order cross-connections



Derivation of repleg

Atomic cross-connection Topology	Path Protection Leg Role	Leg Status	Reported Legpair (*repleg*)

point-to-point	n.a.	1-way	1WAY
point-to-point	n.a.	2-way	2WAY
path-protected	W	1-way	1WAYPSW
path-protected	P	1-way	1WAYPSP

Detailed behavior description

The **RTRV-CRS** command allows the retrieval of existing cross-connection according to the following characteristics:

- rate (via the mandatory *modifier* parameter)
- input or output tributary (via the optional/mandatory [in case *ppgname* is omitted] *in_aid* parameter).

For the rate and input/output tributary, either concrete value or the ranging via *all* (and *vall* for the port-part of the *aid* in case of using the GE***n***, OPTG5, ES64U and *usall* for the slot-part of the *aid*) can be used.

The *ppgname* is applicable for high-order cross-connections and is allowed for compatibility reasons only. The parameter *ppgname* is not evaluated during command processing, i.e. the command returns all cross-connections with the specified rate.

The use of *in_aid* AND *ppgname* in one command request is not allowed; i.e. their use is mutually exclusive.

Number of atomic cross-connections in the response

In case the given set of input parameter does not match with any existing cross-connection(s), the command is accepted BUT no information is returned i.e. the output contains the empty string "".

In case the given set of input parameters matches to a 1way or 2way cross-connection, a single entry will be returned.

In case the given set of input parameters matches to several cross-connections, then one entry per matching cross-connection will be returned.

In case the given set of input parameters matches to one or more pathprotected cross-connection up to two entries will be returned for each path protected cross-connection (one entry for the worker-leg and the second one for the protection-leg if the output tributary has been specified as in_aid parameter, otherwise only the addressed leg). This implies that only the legs meeting the characteristics of the request will be in the reply in case of path-protected XC. The NE will only add matching legs which form together with the reported legs a path-protected XC topology.

In the following, examples are given for the number of cross-connections (xc's) to be returned while tributaries A, B, C, D are cross-connected (disconnected if not mentioned in a specific sceanrio):

- A to B 1WAY1100
- A to B 1WAY + B to A 1WAY2200 (not converted to 2WAY)
- A to B 1WAY + B to C 1WAY1210
- A to B 2WAY1100
- B to A and C to A 1WAYPROT2110
- B to A and C to A 1WAYPROT + A to B 1WAY3210
- B to A and C to A 1WAYPROT + A to B 1WAY + A to C 1 WAY4220
- B to A and C to A 1WAYPROT + A to D 1WAY3111
- B to A and C to A 1WAYPROT + A to B 1WAY + A to D 1WAY4211

AID boundary independence

The system shall allow to retrieve a higher order cross-connection using the AID of any tributary within the AID boundaries determined by the cross-connection rate.

Example: A STS-3c cross-connection has been established at the tributary 1-1-#-#-2-1-4.

This cross-connection can be retrieved with one of the AIDs

- 1-1-#-#-2-1-4
- 1-1-#-#-2-1-5
- 1-1-#-#-2-1-6

AID boundary and rate dependency for LOVC3 cross-connection retrieval

The system shall return a response with the existing LOVC3 cross-connection if the specified LO tributary is within the boundaries of an existing LOVC3 cross-connection and one of the modifiers LOVC3 or ALL is used in the command.

If other modifiers are used or no LOVC3 cross-connection exists at the tributary, then no LOVC3 cross-connection is returned with the response. If no other LO cross-connection exists at the specified tributary, the response will be empty.

If a field of the specified LO tributary AID is set to a value outside the allowed range for this field, then the command is denied. In this case, the generic validation rule for unknown AID sall be applied.

Identification of VCG tributaries

The system shall allow to retrieve cross-connections at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

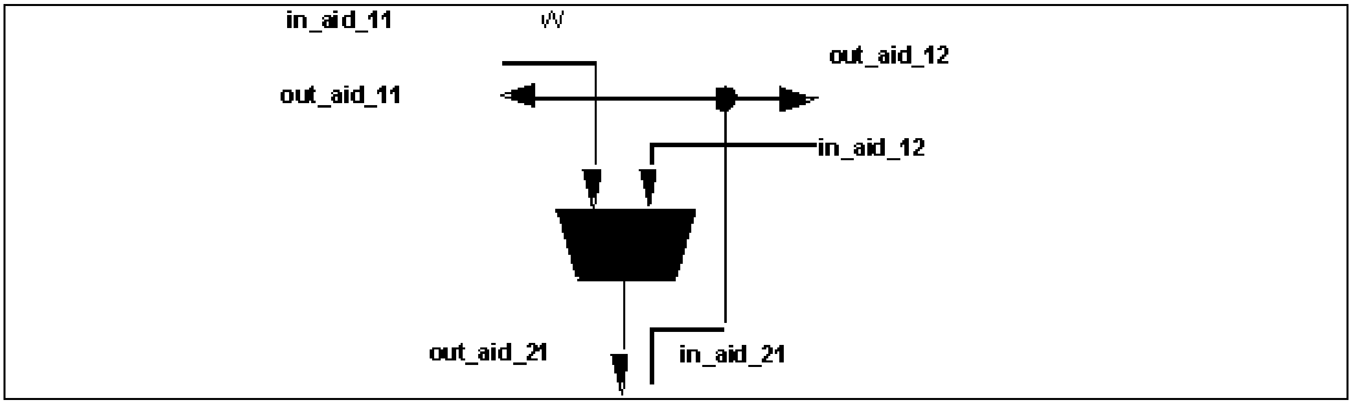
Update affected entities

The **RTRV-CRS** command does not generate a **REPT DBCHG** message.

Example input/output

Example 1

Suppose the following compound cross-connection, as used in UPSR application, is given:



RTRV-CRS-all:UNITE-NE:1-1-#-#-all-all-all:123456; will then output (the blank lines are for better reading):

in_aid_11,out_aid_21:vc4,,in_aid_12:lpblstat=NO,omode=NORM,repleg=1WAYPSW, xcappl=20,xcnum=123456789012;

in_aid_12,out_aid_21:vc4,,in_aid_11:lpblstat=NO,omode=NORM,repleg=1WAYPSP, xcappl=20,xcnum=123456789012;

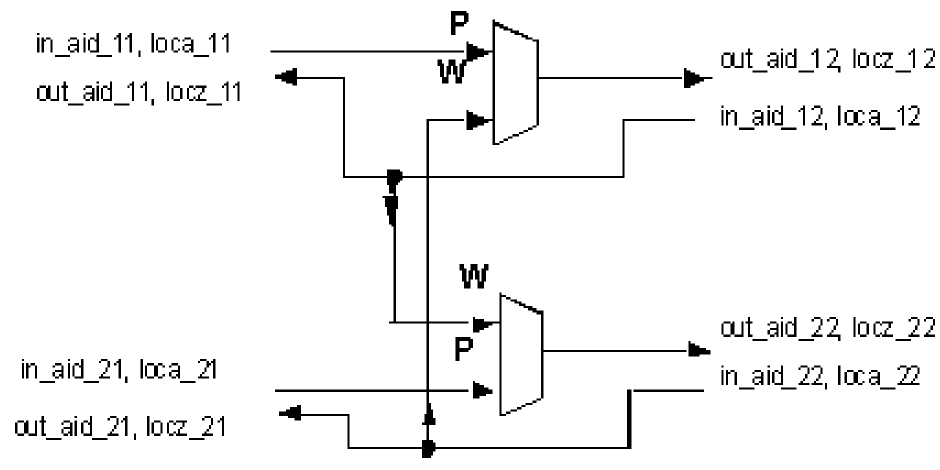
in_aid-21,out_aid_11:vc4:omode=NORM,repleg=1WAY,xcappl=20,xc-num=123456789012;

in_aid-21,out_aid_12:vc4:omode=NORM,repleg=1WAY,xcappl=20,xc-num=123456789012;

The upper 2 outputs specify the path-protected cross-connection, consisting of 2 legs. The lower 2 outputs specify the 2 legs of the bridge.

Example 2

Suppose the following compound cross-connection configuration is given:



A ring interworking with drop and continue for a BLSR with primary nodes in the same NE. RTRV-CRS-all:UNITE-NE:1-1-#-#-all-all-all:123456; will then output (the blank lines are for better reading):

in_aid_11,out_aid_12:vc4,,in_aid_22:loca=loc_a11,locz=locz_12,lpblstat=NO, omode=NORM,repleg=1WAYPSP,xcappl=41,xcnum=011031111222;

in_aid_22,out_aid_12:vc4,,in_aid_11:loca=22,locz_12,lpbkstat=NO,omode=NORM, repleg=1WAYPSW,xcappl=41,xcnum=011031111222;

in_aid_12,out_aid_11:vc4:loca=loca_12,locz=locz_11,omode=NORM,repleg=1WAY, xcappl=41,xcnum=011031111222;

in_aid_12,out_aid_22:vc4,,in_aid_21:loca=loc_a12,locz=locz_22,lpblstat=NO, omode=NORM,repleg=1WAYPSW,xcappl=41,xcnum=011031111222;

in_aid_21,out_aid_22:vc4,,in_aid_12:loca=21,locz_22,lpbkstat=NO,omode=NORM, repleg=1WAYPSP,xcappl=41,xcnum=011031111222;

in_aid_22,out_aid_21:vc4:loca=loca_22,locz=locz_21,omode=NORM,repleg=1WAY, xcappl=41,xcnum=011031111222;

The first 3 outputs specify the upper part of the configuration while the last 3 outputs specify the lower part.

Example 3

Suppose no cross-connections do exist, or the input data do not match with existing cross-connections, then the output would be an empty string: ″″.

RTRV-CRS for a cross-connection established via ENT-CRS-VC4 with a specified in_aid:

```
RTRV-CRS-VC4:LT-UNITE-123:1-1-#-#-1-1-1:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1-1,1-1-#-#-2-1-1:VC4:\
    lpbkstat=NO,omode=NORM,repleg=2WAY,rtnomode=NORM,\
    rtnxcappl=20,rtnxcnum=123456789012,xconcls=TRADITIONAL,\
    xcappl=20,xcnum=123456789012″
;
```

RTRV-CRS for a cross-connection established via ENT-CRS-STS1 at a BLSR port where the Automatic Squelch Map Calculation is enabled:

```
RTRV-CRS-STS1:LT-UNITE-123:1-1-#-#-1-1-1:123456;

   LT-UNITE-123 05-05-31 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1-1,1-1-#-#-2-1-1:STS1:\
    loca=\″LT-UNITE-124\″,locastate=AUTO-VALID,\
    locz=\″LT-UNITE-124\″,loczstate=AUTO-VALID,lpbkstat=NO,\
    omode=NORM,repleg=2WAY,rtnomode=NORM,rtnxcappl=20,\
    rtnxcnum=123456789012,xconcls=TRADITIONAL,xcappl=20,\
    xcnum=123456789012″
;
```

RTRV-CRS for a cross-connection established via ENT-PATH with a specified in_aid:

```
RTRV-CRS-VC4:LT-UNITE-123:1-1-#-#-1-1-1:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1-1,1-1-#-#-2-1-1:VC4:\
    lpbkstat=NO,omode=NORM,repleg=2WAY,rtnomode=NORM,\
    rtnxcappl=0,rtnxcnum=000000000000,xconcls=NN,xcappl=0,\
    xcnum=000000000000″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-CRS** command.

Related TL1 messages

- "DLT-CRS: Delete cross-connection" (p. 3-145)
- "ED-CRS: Edit cross-connection" (p. 3-292)
- "ENT-CRS: Enter cross-connection" (p. 3-1464)

References

GR-199-CORE

RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection

The **RTRV-CRS-FECOM** command can be initiated by the users to retrieve the established transparent DCC cross-connections.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-CRS-FECOM***:tid:[port_aid][,dcctype]:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **port_aid** – DCC port
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel which is involved in a transparent DCC cross-connection. If this field is omitted, the report will include the DCC parameters of all ports belonging to the NE and being provisioned with transparent DCC.
   Type: AID
   Range: SPortAID, the keyword **all** is allowed on slot and port level.
4. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS transparent DCC cross-connection parameters of the OP***n*** port is reported by the command. If this parameter is omitted, LINE and SECTION DCC parameters will be considered for the output.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.

Output format

In response to a valid RTRV-CRS-FECOM command, the following output is returned:

```
    sid date time
 M  ctag COMPLD
    ″port_aid1,port_aid2,dcctype″
      .....
    ″port_aid1,port_aid2,dcctype″
;
```

The output will not be sorted.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Important! The NE responses partially with duplicate lines, if the user retrieves transparent DCC connections by specifing an AID range. The NE lists every port with transparent DCC connections in this AID range. Thus, if both connected DCC ports are in the range, the connection is listed two times. The duplicate lines should be ignored.

Output parameters

Command parameters

1. **port_aid1** – DCC port 1
   This parameter indicates one (first) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID
   Range: SPortAID. The keyword **all** is allowed not.
2. **port_aid2** – DCC port 2
   This parameter indicates one (second) optical port AID of an OP***n*** circuit pack which is used together with dcctype to identify the DCC channel which is involved in a transparent DCC cross-connection.
   Type: AID.
   Range: SPortAID. The keyword **all** is allowed not.
3. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS transparent DCC cross-connection parameters of the OP***n*** port is reported by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.

Detailed behavior description

The System supports only bidirectional transparent DCC cross-connections. Each reported transparent DCC cross-connection is bidirectional.

The command reports all transparent DCC cross-connections. That means only DCC channels are shown with *dccstat* = TRANSPARENT or *scn_dccstat* = TRANSPARENT.

If an AID range is specified, then the DCC cross-connections are reported for all ports in the AID range, i.e. a DCC cross-connection will be reported twice, if both involved ports are in the AID range.

Update affected entities

The **RTRV-CRS-FECOM** command does not generate a REPT DBCHG message.

Example input/output

The following example shows a RTRV-CRS-FECOM command there two transparent DCC cross-connections are reported one for *dcctype* = SECTION and one for *dcctype* = LINE:

```
RTRV-CRS-FECOM:LUCENT-UNITE-NE:1-1-#-#-2-1:12345;
    NE-UNITE 00-12-24 11:00:00
 M  123456 COMPLD
    "1-1-#-#-1-1,1-1-#-#-2-1,LINE"
    "1-1-#-#-1-1,1-1-#-#-2-1,SECTION"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-CRS-FECOM** command.

Related TL1 messages

- "ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
- "DLT-CRS-FECOM: Delete transparent DCC cross-connection" (p. 3-156)
- "RTRV-FECOM: Retrieve DCC related parameters" (p. 3-2407)
- "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)

RTRV-EC1: Retrieve EC1 port

The **`RTRV-EC1`** command is used to retrieve all parameters related to the electrical EC1 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

`RTRV-EC1`:*tid:aid:ctag;*

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   Access Identifier of a EC1 port.
   Type: AID
   Range: SPortAID of EC1 port. The keyword **`all`** is allowed up to shelf level.
3. **`ctag`** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

In response to a valid RTRV-EC1 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **`RTRV-HDR`** command.

Output parameters

Command parameters

1. **`aid`** – access identifier
   Type: AID
   Range: EC1 port AID

Spec block parameters

1. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signal.
   Type: numeric value set
   Range: 11, (1 x STS1).
2. **`eif`** – Electrical Interface Type
   The parameter eif indicates the type of input the port has been provisioned to receive.
   Type: Alphanumeric string.
   Range: EC1
3. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the multiplex section/line.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default)
4. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message “do not use” will be transmitted from that optical interface.
   Type: Alphanumeric value set.

   Range:
   - ENABLE: (set a forced DUS/DNU)
   - DISABLe (default): (clear a forced DUS/DNU)
5. **`fsflpbkstat`** – Far Side Facility Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*. A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback exististing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time
   Type: Alphanumeric value set.
   Range: YES, NO (default)
6. **`inputsig`** – Tributary Input Signal Rate List
   This parameter is a list of STSn rates across the port’s bandwidth.
   Type: numeric value set.
   Range: 11 (default), (1 x STS1).

7. **`istd`** – Interface Standard
Interface standard of the synchronous EC1 interface.
Type: Alphanumeric value set.
Range: SONET
8. **`lbo`** – Line Build Out
This parameter defines the cable length applicable for the EC1 port.
Type: Alphanumeric value set.
Range:
   - SHORT (default): 0 … 68.5 m (0 … 225 ft)
   - LONG: 68.5 … 137 m (225 … 450 ft).
9. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
   - YES
   - NO (default).
10. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
11. **`pmmsfeses`** – Performance Monitoring Line Far-End SES Threshold
This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
Type: Alphanumeric value set.
Range: 1 – 64000; default = 51
12. **`pmmsneses`** – Performance Monitoring Line Near-End SES Threshold
This parameter describes the Line Far-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
Type: numeric value set
Range: 1 – 64000; default = 51
13. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the *pmode* state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range: AUTO (default): indicates that, when a signal is detected at the port, then alarm monitoring will start. MON: indicates that alarm monitoring is taking place on the port. NMON: means that alarm monitoring is not occurring on the port.

14. **`pmrsneses`** – Performance Monitoring Section Near-End SES Threshold
    This parameter describes the Section Near-End SES threshold value. It indicates the number of code violations per second to declare a second as severly errored.
    Type: numeric value set.
    Range: 1 … 64000
    Default: 52
15. **`portClass`** – Port Class
    This identifies the port class of the AID.
    Type: Alphanumeric value set.
    Range: TRADITIONAL
16. **`portControl`** – Port Control
    This identifies the administrative mode of the EC1 port. This parameter is not relevant for EC1 ports, because the portClass is always TRADITIONAL. The parameter can only be retrieved but not changed.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
17. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: EC1
18. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when sdsfmode=POISSON.
    Type: numeric value set.
    Range: (as exponents of 10): -5 -6 (default), -7, -8, -9
19. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.

    Range:

    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
20. **`sfthr`** – Signal Failure Threshold
    Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when sdsfmode=POISSON
    Type: numeric value set
    Range: (as exponents of 10): -3 efault), -4, -5
21. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string
    Range: String with maximum of 24 characters
    Default: DEFAULT0

22. **`sonet`** – SONET ASAP
    SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports of corresponding line rate (see RTRV-STM4, output parameter sdhpn) to be compatible with current generation applications.
    Type: Alphanumeric string
    Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
    Default: DEFAULT
23. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    RANGE: ENABLE, DISABLE (default)
24. **`strc`** – Value of incoming J0
    This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If strcst indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
    Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D)
    Range: 1 byte string
25. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc).
    Type: Numeric string.
    Range: 1 (default) for specific repeating byte (SONET)
26. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted.
    Type: Hexadecimal digits
    Range: 1 byte string. One byte is expressed as two hexadecimal digits (example value: 6D).
    Default: 01Hex
27. **`strcst`** – Status of incoming J0
    This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
    Type: Alphanumeric value set.

    Range:

    - NORMAL: strc contains a valid value
    - UNAVAILABLE: no value for strc could be obtained by the system
    - AIS: no value for strc due to an SF condition.

28. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout).
    Type: Numeric string.
    Range: 1 (default) for specific repeating byte (SONET)
29. **`trbmd`** – Tributary mode
    The parameter trbmd indicates the tributary mode of the port. In the case of EC1 port, the only possible value is FIXED.
    Type: Alphanumeric value set.
    Range: FIXED
30. **`unequiposig`** – Unequipped Output Signal List
    Tributary Unequipped Output Signal Rate List. This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection.
    Type: numeric value set
    Range: 11 (default), 1 xSTS1)

Detailed behavior description

The command shall retrieve the parameters as described above. The command **`RTRV-EC1`** do not generate a REPT DBCHG message.

Example input/output

The following example shows the successful execution of a RTRV-EC1 command. Not all output parameters are listed (missing ones indicated by …).

```
RTRV-EC1:LT-UNITE:1-1-#-#-1-36:123456;

   LT-UNITE 04-12-23 08:00:00
M  123456 COMPLD
   "1-1-#-#-1-36:eif=EC1,ptype=EC1,sfthr=-3,sdthr=-6, pmode=MON,.....
      pmmsneses=51,pmrsneses=52"
;
```

Error responses

See definition in "RTRV-HDR: Retrieve header" (p. 3-2417) command error responses section for general validation rules. The error responses listed there also apply to the RTRV-EC1 command.

Validation rule: Check CCB against *ptype* and *eif*

Check wether the CCB fits to *ptype* and *eif*.

Error code: `IENE – Input, Entity Not Exists`

Related TL1 messages

- "ED-EC1: Edit EC1 port" (p. 3-317)
- "ED-EC1 notification" (p. 3-325)
- "ENT-EC1 notification" (p. 3-1515)
- "DLT-EC1 notification" (p. 3-163)

References

GR-199-CORE (RTRV-*rr*)

RTRV-ELDR: Retrieve ONNS external link data repository

The **RTRV-ELDR** command requests information of the ONNS ELDR data repository, which contains ONNS information about external links to associated neighbor domains.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: NA

Abortable: No.

Input format

The **RTRV-ELDR** command has an input parameter *dataType* that defines what type of information is expected in the *componentID* field that determines what portion of the data repository records is retrieved.

Command syntax: **RTRV-ELDR**:*srcTid:srcAid:ctag::dataType[,componentID];*

Description

The **RTRV-ELDR** command retrieves information stored in the external link data repository that is in RAM Memory. The **RTRV-ELDR** command operates on the current, active-memory-resident, ONNS data. The NVM data from NNDM is used during initialization to recover the ONNS local external link data repository. Data from other ONNS nodes is acquired by the flooding and exchange of LSAs.

Retrieving ALL external link data repository records

When the *dataType* = ALL, Network Navigator shall respond to a request to retrieve its external link data repository by sending a complete report of the contents of the data repository specified. The request shall cause the ONNS node to send all information in the specified data repository. When the external link data repository is retrieved, each TL1 response message contains a record with all of the data relevant to an external link or LSA.

Retrieving external link data repository records for any ASTN node ID

The command **RTRV-ELDR** with *dataType* = LSAID and *componentID* = ASTNNodeID shall retrieve the ELDR LSA records that relate to all of the external links from the specified node with the ASTNNodeID to adjacent domains. This ASTNNodeID can be the address of any node in the ONNS domain.

REPT PTHCHG or REPT NNCHG is not generated

The **RTRV-ELDR** command shall not generate a REPT PTHCHG or REPT NNCHG notification.

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE.
   See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn (default).
3. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″).
   It should be a unique value for each new command.
4. **`dataType`**
   This parameter specifies, which part of the ELDR is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range:
   - *ALL*: retrieves the entire external link data repository in RAM
   - *LSAID*: retrieves the LSA in ELDR related to the ASTN node ID, *componentID* must be an *ASTNnodeID*
5. **`componentID`**
   componentID depends on the previous *dataType* parameter and is one of:
   - not represented when *datatype* = ALL
   - *nodeID*: when *datatype* = LSAID
     Description: Identifier of an ASTN node
     Type: Four dot separated numeric string

Range: 0 … 255, for each numeric string

Output format

The external link data repository record structure

The complete ELDR of one node consists of LSARecords for all nodes with external links (including the reporting node itself) within the same ONNS domain. Within a LSARecord all external links to neighbor domains are kept in LinkRecords (one record for each neighbor domain relation).

External link data repository (ELDR) without data

An addressed node without data in the ELDR shall response only the HeaderRecord with *NumberOfLSAs* = 0. The external link data repository record is composed of an *ELDR_HeaderRecord* and multiple *LSARecords* which are described below:

ELDR_Record ::=

<ELDR_HeaderRecord>

<LSARecord_1>

…

<LSARecord_n>

ELDR_HeaderRecord ::=

eldrID,*dataType[=componentID]*,*srcNodeASTNNodeID*,*NumberOfLSAs= LSA_Count*

LSARecord ::=

<LSAHeaderRecord>

<LSARecord_1>

…

<LSARecord_n>

LSAHeaderRecord ::=

LSAAdvertisedRouter=*origNodeASTNNodeID*,LinkCount=*linkCount*

LinkRecord ::=

LocalNodeID=*localAdvNodeID*,RemoteNodeID=*remoteAdvNodeID*,LocalLinkID=

localLinkID,RemoteLinkID=*remoteLinkID*,*remoteTID*,RemoteRAID=*remoteRAID*,

SRLG=*srlgList*,LinkMetric=*adminCost*,LinkResourceClass=*linkResCls*,

LinkProtType=*protType*,Bandwidth=*bandwidthList*

Output format with normal data retrieve completion report for dataType=ALL

If the **RTRV-ELDR** request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″eldrID,ALL,srcNodeASTNNodeID,NumberOfLSAs=LSA_Count″
   ″LSAHeaderRecord_1:
    LinkRecord_1_1″
   ...
   ″LSAHeaderRecord_n:
    LinkRecord_n_1,
    LinkRecord_n_2″
>
   srcTid date time
M  ctag COMPLD
   ″LSAHeaderRecord_n+1:
    LinkRecord_n+1_1″
   ...
   ″LSAHeaderRecord_m:
     LinkRecord_m_1,
   ...
    LinkRecord_m_n″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the OUTPUT PARAMETERS section of the **RTRV-PATH** command.

ELDR_HeaderRecord

1. **eldrID**
   ONNS Data Repository dataID to be returned
   Type: Alphanumeric value set.
   Range: ELDRexternal link data repository in RAM
   Default: ELDR
2. **dataType**
   This parameter specifies, which part of the ELDR is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range: ALL retrieves the entire external link data repository in RAM. LSAID retrieves the LSA in ELDR related to the ASTN node ID, *componentID* must be an *ASTNnodeID*.
3. **componentID**

componentID depends on the previous *dataType* parameter and is one of:

- not represented when *datatype* = ALL
- *nodeID*: when *datatype* = LSAID
  Description: Identifier of an ASTN node.
  Type: Four dot separated numeric string
  Range: 0 … 255, for each numeric string
  Note: If *dataType* = ALL, *componentID* and the "=" sign are not reported.

4. **`srcNodeASTNNodeID`** – Reporting Node ID
   ASTN node ID of the ONNS node replying to this command.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0 … 255
5. **`LSACount`**
   This parameter indicates the number of LSA records in the report. Also equals number of NE nodes in the network for *dataType* = ALL.
   Type: Integer
   Range:
   - >0: number of LSAs with external link records
   - 0: no ENNI port(s) are defined at this node or within the network

LSARecord

For each reported node of the topology one LSARecord is reported.

1. **`LSAHeaderRecord`**
   All parameters in the LSAHeaderRecord describe properties of the *origNodeASTNNodeID*.
   origNodeASTNNodeID
   Description: The ASTN node ID of the advertiser of the LSA with the external link data following the LSAHeaderRecord
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
   linkCount
   Description: Number of reported links of *origNodeASTNNodeID* towards other nodes in the network.
   Type: Integer
   Range: >0
2. **`LinkRecord`**
   For each external domain neighborship of the *origNodeASTNNodeID* one LinkRecord is reported.
   localAdvNodeID
   Node ID of own domain for level 1 Routing
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
   remoteAdvNodeID
   Corresponding Node ID of neighbor domain for level 1 Routing.

Type: four dot separated numeric string
Range: 0 … 255, for each numeric string

localLinkID

Link ID from own domain for level 1 Routing.
Type: Numeric string.
Range: 0..42946967295

remoteLinkID

Corresponding Link ID from neighbor domain for level 1 Routing.
Type: Numeric string.
Range: 0..42946967295

remoteTID

The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty string (default).
Range: Please refer to the TID definition in the RTRV-HDR description.

remRAID

The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: four dot separated numeric string
Range: 0 … 255, for each numeric string
Default: 0.0.0.0

srlgList

Shared Risk Link Group Identifier list.
See also parameter *srlgList* in RTRV-rr.
The max. number of identities that can be provisioned per port may be release dependent.
Type: Up to ten &-separated Integers
Examples:
198&233&0&0&0&0&0&0&0&0
10&20&30&40&0&0&0&=&0&0
89&1000&4&10&20&31&111&231&300&777
Range: 0 … 65535
Default. 0 (means no SRLG provisioned)
SRLG list is not applicable for EDGE and UNI ports and is set to the default value.

adminCost – Administrative Cost
Provisioned administrative cost of the port connection when used in a connection path. See also parameter *adminCost* in RTRV-rr.
Type: Integer
Range: 1 (default) … 65535
Administrative cost parameter is not applicable for EDGE and UNI ports and is set to the default value.

linkResCls
The *linkResCls* globally classifies routing purposes the link of the corresponding port is assigned to.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF

linkProtType
The *linkProtType* globally classifies routing purposes the link of the corresponding port is assigned to.
Type: Alphanumeric value set.
Range:

- 0x1 (unprotected)
- 1+1-W (1+1 MSP)

bandwidthList
The *bandwidthList* indicates the available bandwidth per signal rate.
Type: Character string.
Range:
Bandwidth List: five dash separated numeric strings representing STS1 #, STS3c #, STS12c #, STS48c #, STS192c # with:

- STS1 #0..4097
- STS3c #0..1023
- STS12c #0..255
- STS48c #0..63
- STS192c #0..15

Example for an available bandwidth of 2 STS48c and a defined minimal XC STS3c: 0-32-8-2-0.

Example input/output

The **RTRV-ELDR** command with *dataType* = ALL produces this example output result:

```
RTRV-ELDR:LT-ONNS-61::123456::ALL;

LT-ONNS-61 2006-10-04 09:45:35
M  123456 RTRV
”ELDR,ALL,135.13.44.61,NumberOfLSAs=3”
”LSAAdvertisedRouter=135.13.44.61,LinkCount=1”
      ”LocalNodeID=128.32.14.32,RemoteNodeID=148.43.56.95,
LocalLinkID=118.23.95.83,RemoteLinkId=184.71.75.23,LT-ONNS-61,
```

```
RemoteRAIID=139.38.28.84,SRLG=0&0&0&0&0&0&0&0&0&0,LinkMetric=24,
LinkResourceClass=34.32.19.1,LinkProtType=0x1,Bandwidth=0-16-4-1-0"
"LSAAdvertisedRouter=139.38.28.84,LinkCount=2"
      "LocalNodeID=148.43.56.95,RemoteNodeID=128.32.14.32,
LocalLinkID=184.71.75.23,RemoteLinkId=118.23.95.83,LT-ONNS-61,
RemoteRAID=218.3.45.56,SRLG=0&0&0&0&0&0&0&0&0&0,LinkMetric=24,
LinkResourceClass=34.32.19.1,LinkProtType=0x1,Bandwidth=0-16-4-1-0"
"LocalNodeID=148.43.56.95,RemoteNodeID=118.23.59.15,
LocalLinkID=23.95.19.45,RemoteLinkId=123.95.82.4,LT-ONNS-61,
RemoteRAID=118.32.145.65,SRLG=1&34&0&0&0&0&0&0&0&0,LinkMetric=260,
LinkResourceClass=1.28.39.3,LinkProtType=1+1,Bandwidth=0-4-1-0-0"
>
LT-ONNS-61 2006-10-04 09:45:35

M  123456 COMPLD
"LSAAdvertisedRouter=129.34.93.2,LinkCount=1"
      "LocalNodeID=118.23.59.15,RemoteNodeID=148.43.56.95,
LocalLinkID=123.95.82.4,RemoteLinkId=23.95.19.45,LT-ONNS-63,
RemoteRAID=124.45.210.56,SRLG=1&34&0&0&0&0&0&0&0&0,LinkMetric=260,
LinkResourceClass=1.28.39.3,LinkProtType=1+1,Bandwidth=0-4-1-0-0"
;
```

Error responses

Invalid *componentID* parameter

The command is rejected when for the datatype=LSAID the received ASTNNodeID is invalid.

Error code: `E511 – IINO – Input, Invalid Node ID`

Access to the database is not possible I

Access to the database is not possible due to internal problems.

Error code:

`E002 – SICE – Status, Inter-Processor Communication Error caused cmd failure`

Access to the database is not possible II

Access to the database is not possible due to internal problems.

Error code: `E003 – SNTT – Status, Network Topology database Timed out`

Related TL1 messages

- "RTRV-L1TDR: Retrieve level 1 topology data repository" (p. 3-2446)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)

- "DLT-CONN: Delete network connection" (p. 3-139)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT CONNCHG notification" (p. 3-2104)

RTRV-ENNIMANROUTE: Retrieve all manual routes of the E-NNI node

The **RTRV-ENNIMANROUTE** command is used to retrieve all manual route entries known by the E-NNI node.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ENNIMANROUTE**:*modifier*:*tid*::*ctag*;

Input parameters

Command parameters

1. **modifier**
   The parameter modifier indicates whether the manual route is used to reach a signaling controller (SC) or a routing controller (RC) subnet.
   Type: Alphanumeric value set.
   Range: SC or RC
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Output format

If the RTRV-ENNIMANROUT is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "spec_block"
       ...
   "spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Output parameters

Spec block parameters

1. **`neighborSIGPCIPAddress`** – Neighbor Signaling Controller IP Address
   This parameter is used to indicate the IP address of a neighboring signaling controller that is connected via a LAN interface.
   The term "neighbor" refers to logical neighborship. The neighboring SC is not necessarily a physical neighbor.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
2. **`neighborSIGPCID`** – Neighbor Signaling Controller Identifier
   This parameter is used to indicate the identifier of a neighboring signaling controller that is connected via a LAN interface.
   The IP address and ID of a signaling controller connected via a point-to-point interfaces is provisioned as part of port provisioning.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *SC* and must not be present if the command modifier has the value *RC*.
3. **`neighborRCIPAddress`** – Neighbor Route Controller IP Address
   This parameter is used to indicate the IP address of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.

4. **`neighborRCIPSubnetMask`** – Neighbor Route Controller IP Subnet Mask
   This parameter is used to indicate the subnet mask of the IP subnet of a neighboring route controller.
   The IP subnet that can be reached via the manual route is described by the tuple neighborRCIPAddress + neighborRCIPSubnetMask.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   Optional parameter: This parameter must be present if the command modifier has the value *RC* and must not be present if the command modifier has the value *SC*.
5. **`lan_nh_ifipad`** – Next Hop Interface IP Address for LAN Interfaces
   This parameter is used to indicate the interface IP address of the next hop of a manual route using a LAN. This parameter is insignificant for point-to-point interfaces.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   The values 0.0.0.0, 1.1.1.1 and 127.0.0.1 are not allowed.
   Optional parameter: This parameter must be present if the interface *if_type* is a LAN. It must not be present, if the interface *if_type* is a point-to-point interface.
6. **`if_id`** – interface id
   This parameter is used to indicate the interface associated with the manual route.
   Type: AID
   Range: SPortAID or LanPort AID (the keyword “all” is not used)
7. **`dcctype`** – DCC type
   This parameter is used to further specify the DCC type, in case the *if_id* refers to a DCC channel.
   Type: Alphanumeric value set.
   Range: LINE or SECTION
   Optional parameter: This parameter must be present if the interface *if_id* is a DCC. It must not be present, if the interface *if_id* is no DCC.
8. **`if_type`** – interface type
   This parameter is used to indicate whether the interface with the interface id if_id is a point-to-point interface (e.g. a DCC) or a point to multi-point interface, i.e. a LAN.
   Type: Alphanumeric value set.
   Range: P2P or LAN

Detailed behavior description

The **`RTRV-ENNIMANROUTE`** command shall retrieve manual routes of the E-NNI node. The SC list of routes also includes the manual routes that are implicitely created when setting the neighbor SC on a point-to-point interface (via the edit port commands).

Example input/output

The following example shows the result of RTRV-ENNIMANROUTE-SC:

```
ENT-MANROUTE-LAN:LT-UNITE:1-1-lan2:123456:::ip_ad=135.168.90.4,ip_ad_
      uni=135.168.90.4,ip_snm=255.255.0.0,osinode=uninode;

   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
;
```

The following example shows the result of RTRV-ENNIMANROUTE-RC:

```
ENT-MANROUTE-LAN:LT-UNITE:1-1-lan2:123456:::ip_ad=135.168.90.4,ip_ad_
      uni=135.168.90.4,ip_snm=255.255.0.0,osinode=uninode;

   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ENNIMANROUTE** command.

Validation rule: E-NNI Node must be created

If the E-NNI node is not created (by provisioning of the *sigPCIP_Address* in **ENT-ULS**), then the command is denied.

Error code: `SNVS – Status, Not in Valid State, E-NNI Node not created`

Related TL1 messages

- “ENT-ENNIMANROUTE: Enter manual route to the E-NNI node” (p. 3-1520)
- “DLT-ENNIMANROUTE: Delete manual route from the E-NNI node” (p. 3-164)

RTRV-EPM: Ethernet performance monitoring

The **RTRV-EPM** command is initiated by a user to request the NE to send the current and/or historical Performance Monitoring (PM) data associated with one or more Ethernet ports. For each value in the *emontype* parameter of the **RTRV-EPM** input Parameters section, PM counter values will be stored in the current 15-minute period, the 32 previous 15-minute periods (past 8 hours worth of data), current day, and previous day periods.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-EPM-*modifier*:*tid*:*aid*:*ctag*::[*emontype*], [*monlev*],*tmper*, [*mondat*], [*montm*];

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the Ethernet port on which the **RTRV-EPM** command is to act on.
   Type: Alphanumeric value set.
   Range: ENET, 1GEPL, 10GEPL.
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   Type: AID.
   Range: EPortAID, VCGAID.
   Range is supported up to shelf level or system level; i.e.
   ALL-ALL-#-#-ALL-ALL (range over EPortAID)
   ALL-ALL-#-#-ALL-vALL (range over VCGAID).
4. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417) in the input parameters section.

5. **`emontype`** – Monitored parameter type
   This is the monitored parameter type for which data is requested. If the value is omitted, this parameter defaults to ALL.
   Type: value set.
   Range for modifier ENET: CBR, CBS, PDE.
   Range for modifiers 1GEPL and 10GEPL: EINB, EONB, EINF, EONF, EDFE, EDFC
   The following value is for input monitored type only: ALL.
   Default: ALL.
6. **`monlev`** – Monitored Level
   This parameter provides the level for the monitored PM parameter. The parameter is only evaluated if the value of *montype* parameter is ALL.
   Type: value set.

   Range:

   - 1-UP: Means the NE will report all non-zero data and all data which is marked as non-valid (parameter validity).
   - 0-UP: Means the NE will report zero and all positive data.

   Default: 1-UP.
7. **`tmper`** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.

   Range:

   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.
8. **`mondat`** – Monitored Date
   This requests the beginning date of the interval for which the PM data is to be reported. For both values of tmper parameter, the maximum allowable range of valid dates includes the current and previous one day.
   Type: value set.

   Range:

   - MM-DD: Month-Day. A leading zero (first M or first D) is optional and may be omitted.
   - ALL: Current and previous day.

   Default: Current date
9. **`montm`** – Monitored Time

This parameter specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. The user is able to specify a monitored time in three modes:

a. A monitored time interval (i.e., from hour-minutes to hour-minutes: HOD-MOH&&THOD-TMOH).

b. A single time interval (i.e., the hour and minutes for the appropriate 15 minute data: HOD-MOH).

c. All applicable beginning times for the given *mondat* value (i.e., all).

Where HOD-MOH represents the start of the requested PM retrieval period (hours and minutes) and THOD-TMOH represents the end of the requested PM retrieval period. The value ALL must be used if the *mondat* value provided is ALL.
Type: Character string.

Range:

- HOD-MOH: This specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. This parameter may be omitted or have one of the following values: Hour Of Day- Minute Of Hour, where HOD ranges from 0 to 23 and MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.
- THOD-TMOH: To Hour Of Day-To Minute Of Hour, where THOD ranges from 0 to 23 and TMOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0. Parameter grouping is used when specifying THOD-TMOH. For example, 08-15&&09-00 specifies the HOD-MOH and THOD-TMOH of 08-15 and 09-00, respectively. THOD-TMOH of $$-$$indicates the current Hour Of Day-Minute Of Hour. THOD-TMOH of ##-##indicates the previous 15-min interval.
- ALL: All applicable beginning times for the given *mondat* value.

If the *tmper* value is 1-DAY, the *montm* parameter will not be used.
Default:
If THOD-TMOH is not specified, only the PM data for the retrieval period specified by HOD-MOH will be retrieved.
If HOD-MOH is omitted, the PM period for the current time will be reported.

Output format

Normal completion message

If the NE fully complies with the RTRV-PM request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″aid,aidtype:emontype,monval,[validity],[tca],tmper,[mondat],[montm]″
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Normal completion message – empty reply

If there is no PM data stored within the NE, then the following response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

1. **`aid`** – Access Identifier
   Description: Access identifier.
   Type: AID.
   Range: EPortAID, VCGAID.
2. **`aid`**type – Aid type
   Type: value set.
   Range: ENET
3. **`emontype`** – Monitored parameter type
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417) in the input parameters section.
4. **`monval`** – Monitored Value
   This parameter contains the measured value of the parameter specified in *montype*.
   Type: Integer.
   Range: 0-108000000000000
   The specified range is the maximal range taking all PM parameters into consideration. Specific PM parameters may have a smaller range.
   W.r.t. value range for the specific PM parameters see
   - TCA profile section (if thresholding is applicable for that parameter)
   - PM parameter introduction section (if thresholding is not applicable for that parameter).
5. **`validity`** – validity indicator
   This parameter indicates the validity of the PM data.
   Type: Integer.
   Range: 1: valid; (omitted): non-valid.
6. **`tca`** – Threshold Crossing Alert
   Type: value set.

Range for the current interval:

- in TCA mode TR, that the current value exceeds the provisioned threshold
- in TCA mode TR-RTR, that a *tca* occurred since the beginning of this; interval or that the tca reset condition was not fulfilled in any previous; interval since the last tca occurred

f the value is omitted, it indicates

- in TCA mode TR, that the current value is below the provisioned threshold
- in TCA mode TR-RTR, that no tca occurred since the beginning of this; interval and that the tca reset condition was fulfilled at least in one previous , interval

For previous intervals, the tca parameter indicates the state at the end of the interval.

7. **`tmper`** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.
8. **`mondat`** – Monitored Date
   This parameter specifies the beginning date of the interval for which the PM data is reported. The parameter can be omitted in case of current date.
   Type: value set.
   Range: MM-DD: Month-Day
   Default: Current date
9. **`montm`** – Monitored Time
   This parameter specifies the beginning time of day for the reported pm register. The parameter is optional and may be omitted for the reporting of current bins (registers).
   Type: Character string.
   Range: HOD-MOH: HOD ranges from 0 to 23; MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.

Example input/output

The following example shows the successful completion of a **`RTRV-EPM`** command by the NE:

```
RTRV-EPM-ENET:UNITE-NE:1-1-#-#-01-1:123456::PDE,,15-MIN,01-01,08-00;
   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
   ”1-1-#-#-01-1,ENET:PDE,2,,,15-MIN,01-01,08-00”
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-EPM** command.

Related TL1 messages

- "RTRV-PM: Retrieve performance monitoring" (p. 3-2731)
- "DLT-TCA-PROF: Delete TCA profile" (p. 3-259)
- "ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
- "ENT-TCA-PROF: Enter TCA Profile" (p. 3-1894)
- "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982)

References

GR-833-CORE

RTRV-EPORT: Retrieve Ethernet port

The **RTRV-EPORT** command retrieves the attributes of one or more Ethernet ports.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-EPORT**-*modifier*:*tid*:*aid*:*ctag[:::spec_block]*;

Input parameters

Command parameters

1. **modifier**
   This parameter specifies the type of the Ethernet port at the LAN side of the unit.
   Type: Alphanumeric value set.
   Range:
   - 1GE representing a 1 GE port on a GE1 pack
   - 1GEPL representing a 1 GE port on a GE10PL1 pack
   - 10GEPL representing a 10 GE port on a GE10PL1 pack.
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   This parameter specifies the Ethernet port to be operated on.
   Type: AID
   Range: EPortAID as defined in AID structure. The values
   - vall for Simple_EPortAID
   - usall for Simple_UnivSlotAID
   - all for Simple_UnivSlotAID

- all for Simple_ShelfAID and
- all for BayAID

are accepted.

4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Spec block parameter

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values *all* or *usall* at the appropriate positions of the input AID. For input AIDs not using *all* or *usall* this parameter is not allowed. This implies that with such input AIDs the dynamic parameters are always returned.

   The following output parameters of an Ethernet port are affected:

   - anegst
   - fcmdo
   - macaddr
   - rfltst
   - stmd
   - stpop
   - stpid
   - stprole
   - vss
   - vvidlsd.

   Note that not only those parameters have to be returned which are supported by the related Ethernet pack. Not all Ethernet packs support all listed parameters. For details refer to the detailed specification of those output parameters.
   Type: Alphanumeric value set.
   Range: YES: Report dynamic parameters in the reply as well; NO (default): Do not report dynamic parameters in the reply.

Output format

After receiving the **RTRV-EPORT** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ”aid,modifier::spec_block”
```

```
       ...
       ...
   "aid,modifier::spec_block"
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the Ethernet port to be operated on.
   Type: AID.
   Range: EPortAID as defined in AID structure. The usage of the following values are excluded:
   - all in the BayAID definition
   - all in the Simple_ShelfAID definition
   - usall in the Simple_UnivSlotAID definition
   - all in the Simple_UnivSlotAID definition
   - all in the Simple_EPortAID definition.
2. **modifier**
   This parameter specifies the type of the Ethernet port at the LAN side of the unit.
   Type: Alphanumeric value set.
   Range: 1GE representing a 1 GE port on a GE1 pack, 1GEPL representing a 1 GE port on a GE10PL1 pack, 10GEPL representing a 10 GE ports on a GE10PL1 pack.

Spec block parameters

1. **anegst** – Auto Negotiation Status
   This parameter reports the result of the autonegotiation process (e.g. flow control) of this port with its link partner.
   This parameter is not supported if the input parameter *modifier* = 10GEPL. If the parameter *anegst* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - ANER, for auto negotiation error
   - CFG, for configuring
   - CPL, for complete

- UNAVAILABLE, for not available/unavailable
- DISABLE, for being disabled.

Restrictions: The value DISABLE is not returned if the input parameter *modifier* = 1GE. The values ANER, CFG, CPL and UNAVAILABLE is not returned if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = DISABLE.

2. **`autoneg`** – Auto Negotiation Mode
   This parameter controls auto-negotiation process to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE or 10GEPL. If the parameter *autoneg* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: ENABLE (default); DISABLE.
3. **`cnprole`** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer LAN port or a network LAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter *cnprole* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: CUSTOMER (default), for a customer LAN port; NETWORK, for a network LAN port.
4. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter *dftpri* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
5. **`dftvlan`** – Default VLAN Identifier
   This parameter specifies the value of the VLAN identifier which is applied to frames (tagging) at LAN ingress (i.e. from LAN into the Ethernet-port). This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter *dftvlan* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- 1, 2, 3, … 4092, 4093
- UNAVAILABLE (default).

To express the state UNAVAILABLE system internally the invalid VLAN ID 4094 is used.

6. **droppause** – Drop Incoming Pause Frames
This parameter controls whether incoming pause frames are dropped (value = ENABLE) or not (value = DISABLE). This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter droppause is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE
Default: DEFAULT.
7. **enetpn** – Ethernet Alarm Severity Assignment Profile Name
This parameter points to an ASAP profile name of type ENET for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: 1 … 24 characters length.
Default: DEFAULT.
8. **enettca** – Ethernet TCA Profile Name
This parameter specifies the name of the threshold crossing alert (TCA) profile used for Ethernet ports.
Type: Alphanumeric string.
Range: 1 … 24 characters length.
Default: DEFAULT0.
9. **epmode** – Ethernet Port Monitoring Mode
This parameter specifies the monitoring mode of the Ethernet port. Alarms for this port will be reported if in state MON otherwise not. The configuration of epmode will change from AUTO to MON if a valid signal is applied to that port.
Type: Alphanumeric value set.
Range:
   - AUTO (default), for automatic
   - MON, for monitored
   - NMON, for not monitored.
10. **eppm** – Ethernet Port Performance Monitoring
This parameter specifies whether the performance monitoring is enabled or disabled for this port.
Type: Alphanumeric value set.
Range:
    - DISABLE (default)
    - ENABLE

11. **fcmd** – Flow Control Mode
This parameter specifies the configuration for the pause operation of this port. If the input parameter *modifier* = 1GE, then the parameter *fcmd* is supported only in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
Type: Alphanumeric value set.
Range:
   - AUTO, for auto-negotiate
   - DISABLE, for pause mode bidirectionally disabled
   - ASYMTOPARTNER, for pause mode having receive disabled, transmit enabled
   - ASYMTOLOCAL, for pause mode having transmit disabled, receive enabled
   - SYM, for pause mode bidirectionally enabled.

   Restrictions: If the parameter *fcmd* is not supported, then the parameter is omitted. The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GE.
   The values ASYMTOPARTNER, ASYMTOLOCAL and SYM is not supported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = ENABLE. The value AUTO is notsupported if the input parameter *modifier* = 1GEPL and the parameter *autoneg* = DISABLE.
   The value AUTO is not supported, if the input parameter *modifier* = 10GEPL.
   Possible combinations of the parameters *modifier*, *autoneg*, *vlantagmd* (parameter of the referring circuit pack entity, see "RTRV-EQPT (CP): Retrieve equipment (circuit pack)" (p. 3-2375) for definition) and *fcmd* are as follows:

Input parameter *modifier*	**In-/output parameter *autoneg***	**In-/output parameter *vlantagmd***	**In-/output parameter *fcmd***
1GE	not supported	BCMD	AUTO, DISABLE
		other than BCMD	not supported
1GEPL	ENABLE	not supported	AUTO, DISABLE
	DISABLE		DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM
10GEPL	not supported	not supported	DISABLE, ASYMTOPARTNER, ASYMTOLOCAL, SYM

   Default:
   - AUTO (for *modifier* 1GE or 1GEPL)
   - SYM (for *modifier* 10GEPL).

12. **fcmdo** – Flow Control Mode Operation
This parameter reports the status of the pause operation at this port. If the input parameter modifier = 1GE, then the parameter fcmdo is meaningful only in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
Type: Alphanumeric value set.

Range:

- DISABLE, for Transmit and Receive Disabled
- ENABLE, for Transmit and Receive Enabled
- ENXDISR, for Transmit Enabled, Receive Disabled
- DISXENR, for Transmit Disabled, Receive Enabled
- UNAVAILABLE, for anegst equal ANER or CFG or if the value can not be retrieved.

13. **`fsflpbkstat`** – Far Side Facility Loopback Status
This parameter reports the configuration of the far-side facility loopback of this port. A far-side loopback means that the signal on the outgoing port is looped to the incoming port (Tx -> Rx). For GE1 there is only this type of facility loopback.
Type: Alphanumeric value set.
Range:
    - YES, for enabled loopback
    - NO (default), for disabled loopback.
14. **`macaddr`** – Port MAC Address
This parameter specifies the MAC address of the port.
Type: Alphanumeric value set.
Range: 000000000000 to FFFFFFFFFFFF or UNAVAILABLE if the value cannot be retrieved.
15. **`rcf_htmr`** – Remote Client Fail Hold Off Timer
The parameter specifies the hold-off time for applying the LPT consequent action to disable the laser at the LAN port. The hold-off timer is only applicable if there is a strict one-to-one relationship between the LAN port in question and its related WAN port.
This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_htmr* is not supported, then the parameter is omitted.
Type: numeric string [ms].
Range: 0 … 2000 ms in 100 ms steps
Default: 0.
16. **`rcf_linkdown`** – Link Pass Through Mode
This parameter controls the consequent action (to switch off the laser upon receipt of a client signal fail [CSF] indication) to be enabled or disabled. This parameter is not supported if the input parameter *modifier* = 1GE. If the parameter *rcf_linkdown* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE.
Default: ENABLE.
17. **`rfltst`** – Remote Fault Status
This parameter reports the fault status of the remote link partner.
Type: Alphanumeric value set.

Range for modifiers 1GE and 1GEPL:

- ANER, for auto negotiation error
- LNKFL, for link failure
- UNAVAILABLE, for not available/unavailable
- NOER, for no error
- OFFL, for offline.

Range for modifier 10GEPL:

- REMOTEFLT, for remote fault
- UNAVAILABLE, for not available/unavailable
- NOER, for no error
- LOCALFLT, for local fault.

18. **`rptmode`** – Repeater Mode
    This parameter sets the bridge device for this LAN port into the repeater mode (promiscuous mode), valid for a point to point connection between one LAN and one WAN port. Operating in repeater mode imposes some restrictions. See detailed behavior section.
    If the input parameter *modifier* = 1GE, then the parameter *rptmode* is meaningful only if in case that the bridge operates in backward-compatible mode (*vlantagmd* = BCMD).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter rptmode is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
19. **`servcond`** – Service Condition
    This parameter specifies the service condition for this port. Some configuration possibilities like loopbacks may be dependent on the state of *servcond*.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default)
    - OUTOFSERVICE.
20. **`stdis`** – Spanning Tree Protocol Disable
    This parameter allows to enable or disable the Spanning Tree Protocol for the given port. If the input parameter modifier = 1GE, then this parameter is meaningful only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the

database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL. If the parameter stdis is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- DISABLE, STP is disabled, i.e. the port is not used in the spanning tree protocol
- ENABLE (default), STP is enabled, i.e. the port is used in the spanning tree protocol.

21. **`stmd`** – Spanning Tree Protocol Mode
This parameter returns whether the normal STP according to IEEE802.1D or the rapid spanning tree protocol rSTP according to IEEE802.1w is used on the given port. If the input parameter modifier = 1GE, then this parameter is meaningful only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stmd is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- 802.1D, for the normal ST
- 802.1W, for rSTP
- UNAVAILABLE if the value can not be retrieved.

Default: There is no default value.

22. **`stpcost`** – Spanning Tree Path Cost
The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
If the parameter *stpcost* is not supported, then the parameter is omitted.
Type: Integer.
Range: 1 to 200000000.
Default: 20000.
The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

23. **`stpid`** – Spanning Tree Port Identifier
    The port identifier is a 16 bit parameter consisting of the port priority (high-order bits) and the port number (low-order bits). The port number is assigned by the system to a port and is not provisionable. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK).. For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: 0 to 65535 and UNAVAILABLE if the value can not be retrieved.
24. **`stpop`** – Spanning Tree Protocol Operational
    This parameter indicates whether the Spanning Tree Protocol is operational or not. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK).. For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: YES, operational; NO, not operational, UNAVAILABLE if the value can not be retrieved.
25. **`stpp`** – Spanning Tree Port Priority
    This parameter specifies the priority of the given port for the Spanning Tree Protocol. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.
    The stepping as recommended by the standard (steps of 16 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 240″ is not done by the NE.
    Default: 128.
26. **`stprole`** – Spanning Tree Protocol Role
    This parameter reports the role of the given port in rSTP. If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK) and operates in rSTP mode (stmd = 802.1W). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However,

changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter stpcost is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- ROOT, root
- DESG, designated
- ALT, alternative
- BACKUP, backup
- UNAVAILABLE if the value can not be retrieved.

27. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding. Two thresholding methods are applicable: TR-only. (also known as the implicit reset method), TR-RTR. (also known as the explicit reset method).
Type: Alphanumeric value set.
Range: TR, TRRTR (default).

28. **`vrtswid`** – Virtual Switch Identifier

This parameter is not supported if the input parameter

- *modifier* = 1GEPL, or *modifier* = 10GEPL.

This parameter identifiers the instance of the virtual switch.

- A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
- There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

Type: Integer.
Range: 1 **–** Currently this is the only allowed value.
Default: 1.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or modifier = 10GEPL.
Option: If the parameter vrtswid is not supported, then the parameter is omitted. If not generally omitted, then this parameter is reported only if the virtual switch entity exists (i.e. the VLAN tagging mode *vlantagmd* of the circuit pack the addressed EPort belongs to is set to 802.1TAG).

29. **`vss`** – Spanning Tree Status
This parameter returns the status of the spanning tree protocol for the given port. If the input parameter modifier = 1GE, then this parameter is applicable only if the port is a network port (cnprole = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.

This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
The name of this parameter originally stands for VCG STP status which is somewhat misleading for an EPORT but is kept for consistency reasons with other products. (When using an EPORT as a network port it participates in STP and therefore also has status information.) If the parameter vss is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- FWG, Forwarding
- LRG, Learning
- LIS, Listening (IEEE8102.D/STP only)
- BLK, Blocking (IEEE8102.D/STP only)
- DISF, Disabled due to a failure (IEEE8102.D/STP only)
- DISA, Disabled via provisioning
- BRK, Broken (IEEE8102.w/rSTP only)
- DCD, Discarding (IEEE8102.w/rSTP only)
- UNAVAILABLE if the value can not be retrieved.

30. **vvidls** – Valid VLAN Identifier List (static)
This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using ’&’ and ’&&’ . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (cnprole = NETWORK).
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter vvidls is not supported, then the parameter is omitted.
Type: Alphanumeric string and the character ’&’.

Range:

- Value of each single member (valid VLAN identifier) in the list 1, 2, ..., 4092, 4093
- In case the list has no members (empty list) -1.

The length of the list is limited by the usual system limitations like e.g. TL1 line length and some NE internal restrictions as described below. The number of entries in this list is limited to 64 by some NE-internal restrictions. Ranging and grouping by using the operators ’&’ and ’&&’ has no impact on the number of entries, it only reduces the length of the list being transported to/from CIT/EMS.
Default: -1.

Example 1:
vvidls=-1. The list has no members (empty list).
Example 2:
vvidls=3&5&40&&45. This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.
Example 3:
vvidls=1&&4093 This list has the members 1,2, 3, …, 4092, 4093.

31. **vvidlsd** – Valid VLAN Identifier List (dynamic)
This parameter gives the list of all dynamic VLAN identifiers allowed for this port as gathered autonomously via GVRP. The list is ampersand grouped (& and &&). The vlan identifier of an incoming frame must be a member of this dynamic VLAN Id list or the static VLAN Id list vvidls. Frames tagged by a vlan identifier not member of these lists will be discarded. This parameter is also used in transparent tagging mode to report the list of valid customer tags (CIDs) for network ports (cnprole = NETWORK) collected by GVRP.
This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL. If the parameter vvidlsd is generally not supported, then the parameter is always omitted.
Otherwise, this parameter is reported only if

- the virtual switch entity exists (i.e. the VLAN tagging mode vlantagmd of the slot the addressed EPort belongs to is set to 802.1TAG) AND
- GVRP is enabled on the virtual switch (*gvrpmode* = ENABLED)
- the port is a NETWORK port (*cnprole* = NETWORK).

Type: Alphanumeric string and the character ’&’.
Range:

- Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
- In case the list has no members (empty list) -1.

The length of the list is limited by the usual system limitations like e.g. TL1 line length and some NE internal restrictions as described below. The number of entries in this list is limited to 247 by some NE-internal restrictions. Ranging and grouping by using the operators ’&’ and ’&&’ has no impact on the number of entries, it only reduces the length of the list being transported to/from CIT/EMS.
Default: There is no default value.-1
Example 1:
vvidlsd=-1 The list has no members (empty list).
Example 2:
vvidlsd=3&5&40&&45 This list has the members 3, 5, 40, 41, 42, 43, 44 and 45.

Detailed behavior description

The values of the following parameters are directly requested form the hardware:

- anegst
- fcmdo

- macaddr
- rfltst
- stmd
- stpop
- stpid
- stprole
- vss
- vvidlsd.

Thus the value ″UNAVAILABLE″ is reported, if the CP HW is not available in the system.

Note that not only those parameters have to be returned which are supported by the related Ethernet pack. Not all Ethernet packs support all listed parameters. For details refer to the detailed specification of those output parameters.

Example input/output

Example for a 10GE port of a GE10PL1 pack:

```
RTRV-EPORT-10GEPL:LT-UNITE:1-1-#-#-1-9:123456;
   LT-UNITE 02-05-21 11:11:00
M  123456 COMPLD
   ″1-1-#-#-1-9,10GEPL::droppause=ENABLE,\
    enetpn=DEFAULT,enettca=DEFAULT,epmode=AUTO,eppm=DISABLE,fcmd=SYM,\
    fcmdo=UNAVAILABLE,fsflpbkstat=NO, \
    rcf_htmr=2000,rcf_linkdown=DISABLE, \
    macaddr=112233445566,rfltst=UNAVAILABLE, \
    servcond=INSERVICE,tcamd=TR″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-EPORT** command.

Validation rule: *modifier* consistent with given port-rate

The system shall verify that the modifier parameter is consistent with the rate of the port (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-EPORT** command for applicable combinations.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with existing port rate`

Validation rule: *modifier* consistent with the related circuit pack type

The system shall verify that the modifier parameter is consistent with the related pack type (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-EPORT** command for applicable combinations.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with the related circuit pack type`

Related TL1 messages

- "ED-EPORT: Edit Ethernet port" (p. 3-329)
- "OPR-LPBK: Operate facility loopback" (p. 3-2044)
- "RTRV-LPBK (facility): Retrieve loopback (facility)" (p. 3-2511)
- "RLS-LPBK: Release loopback (facility)" (p. 3-2198)

RTRV-EQPT (bay): Retrieve equipment (bay)

The **RTRV-EQPT** command retrieves information associated with the bay level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-EQPT**:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Description: The AID of the bay/rack.
   Type: AID.
   Range: **all**
   Default: none.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

aid – Access Identifier

Description: The AID of the bay/rack.

Type: AID

Range: BayAID. The keyword **all** is not allowed as output parameter.

Default: 1

Detailed behavior description

The system supplies a normal completion response with the AID of the bay/subrack.

Example input/output

```
RTRV-EQPT:UNITE-NE:all:123456;

   UNITE-NE 07-12-23 08:00:00
M  123456 COMPLD
   "1"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to RTRV-EQPT (bay) command.

References

GR-199-CORE

RTRV-EQPT (CP): Retrieve equipment (circuit pack)

The **RTRV-EQPT** command retrieves the provisioning information associated with a circuit pack. The **RTRV-EQPT** command retrieves the inventory information associated with other replaceable units.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-EQPT**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a circuit pack, the backplane or another replaceable unit (e.g. user panel or fan unit).
   Type: AID
   Range:
   - CpAID ::= UnivSlotAID-cp for universal circuit packs
   - CpAID ::= XCSlotAID-cp for cross-connect circuit packs
   - CpAID ::= CTLSlotAID-cp for controller circuit packs
   - CpAID ::= SlotAID-cp for any circuit pack
   - LoxcAID::=UnivSlotAID-loxc for LOXC packs in universal slots
   - CpAllAID::= UnivSlotAID-cpall for any circuit pack in universal slots
   - CpAllAID::=SlotAID-cpall for any circuit pack in any slot
     The keyword **xcall** is allowed for Simple_XCSlotAID.
     The keyword **ctlall** is allowed for Simple_CTLSlotAID.
     The keyword **all** is allowed for Simple_SlotAID
   - BackplaneAID for backplane. The keyword **all** is not allowed for the backplane AID.
   - UserPanelAID for user panel. The keyword **all** is not allowed for the user panel AID.
   - FanAID for fan. The keyword **fanall** is not allowed for FanAID.

- TimingInterfaceAID for TI Interface board. The keyword **tiall** is allowed for TimingInterfaceAID.
- CiCtlAID for CI-CTL Interface board. The keyword **all** is not allowed for the CiCtl AID
- PowerAID for power interface The keyword **piall** is allowed for PowerAID.
- EciAID for Electrical Connector interface The keyword **eciall** is allowed for EciAID.

The keyword **all** is not allowed on bay/shelf level.

3. **ctag** – Correlation Tag
Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

1. **aid** – Access Identifier
The AID of a circuit pack, the backplane or another replaceable unit (e.g. user panel or fan unit).
Type: AID
Range:
   - CpAID ::= SlotAID-cp for any slot
   - LoxcAID ::= UnivSlotAID loxc for LOXC pack in universal slots
     The keyword **usall** is allowed for Simple_UnivSlotAID.
     The keyword **xcall** is allowed for Simple_XCSlotAID.
     The keyword **ctlall** is allowed for Simple_CTLSlotAID.
     The keyword **all** is allowed for Simple_SlotAID
   - BackplaneAID for backplane. The keyword **all** is not allowed for the backplane AID.
   - UserPanelAID for user panel. The keyword **all** is not allowed for the user panel AID.

- FanAID for fan. The keyword **`fanall`** is not allowed for FanAID.
- TimingInterfaceAID for TI Interface board. The keyword **`tiall`** is allowed for TimingInterfaceAID.
- CiCtlAID for CI-CTL Interface board. The keyword **`all`** is not allowed for the CiCtl AID
- PowerAID for power interface The keyword **`piall`** is allowed for PowerAID.
- EciAID for Electrical Connector interface The keyword **`eciall`** is allowed for EciAID.

Refer to the AID syntax description.

Spec block parameters

1. **`name`** – Circuit Pack Name
   Circuit Pack Name as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *name* parameter is not reported for an LOXCAID.
   Type: value set.
   Range: up to 12 characters.
2. **`qual`** – Circuit Pack Qualifier
   Circuit Pack Qualifier as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, if equipped and supported.
   This parameter is not reported for a CpAID. The *qual* parameter is not reported for an LOXCAID.
   Type: value set.
   Range: up to 7 characters.

   Note: The TL1-representation of the functional qualifier follows the rules as defined in SRD_ConfigurationManagement:

   - alphanumeric characters or the special character ’-’ only
   - special characters like ’/’ and ’.’ to be omitted.
3. **`app`** – Apparatus Code
   Apparatus code as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *app* parameter is not reported for an LOXCAID.
   Type: Alphanumeric string.
   Range: up to 9 characters.

4. **ssn** – Series Number
   Series number as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *ssn* parameter is not reported for an LOXCAID.
   Type: value set.
   Range: up to 6 characters.
   Note: A dash (-) is used instead of a colon (:) for the series number in TL1.
5. **clei** – Common Language Equipment
   Common Language Equipment as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user-panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *clei* parameter is not reported for an LOXCAID.
   Type: value set.
   Range: 10 characters.
6. **eci** – Equipment Catalog Item
   Equipment Catalog Item as stored in the EEPROM of the replaceable unit.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *eci* parameter is not reported for an LOXCAID.
   Type: value set.
   Range: 6 digits.
7. **sln** – Serial Number
   Serial number as stored in the EEPROM on the circuit pack to identify the replaceable unit and indicate the date and place of manufacture.
   This parameter should be reported for the AID for the backplane, AID for the user panel , AID for CI-CTL, AID for TI-Interface, AID for the fan, AID for power interface, AID for ECI, if equipped and supported.
   This parameter is not reported for a CpAID. The *sln* parameter is not reported for an LOXCAID.
   Type: Alphanumeric string.
   Range: 12 characters.

8. **`dcn_ipad`** – DCN IP address for CLI/SNMP
   DCN IP address for CLI/SNMP management of the card.
   This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255.
   The values 0.0.0.0, 1.1.1.1, 127.0.0.1, and 224.0.0.0 to 255.255.255.255 are excluded.
   Default: Empty string.
9. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the circuit pack. This parameter is reported for a CpAID , PowerAID, EciAID, imingInterfaceAID and LoxcAID.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
10. **`eqptprotpn`** – Protected Equipment ASAP Profile Name
    The name of the current ASAP profile of type EQUIP-PROT for the equipment.
    Option: This parameter is only reported for a CpAID, LoxcAID or PowerAID.
    Type: Alphanumeric string.
    Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
    Default: DEFAULT.
11. **`es64ucap`** – ES64U capacity
    Switch capacity of the ES64U pack.
    This parameter is relevant only for circuit packs of type ES64U (that means, *nvmname* = ES64U). The parameter shall be omitted for other packs.
    Type: Value set.

    Range:

    - 0 – ES64U in suspended state, initial value after ES64U CP creation.
    - 20 – ES64U active or member of equipment protection group, capacity 20 Gbit/s.

    Default: 0.
12. **`vlantgmd`** – VLAN Tagging Mode
    VLAN tagging mode of the circuit pack. This parameter is relevant only for circuit packs of type GE1 (i.e. nvmname = GE1).
    Type: Value set.

    Range:

    - BCMD (default), for the backward compatible mode
    - 802.1tag, for the VLAN tag based mode according to IEEE802.1Q.
13. **`loswcap`** – LOXC switch capacity
    Switch capacity of the LOXC pack. This parameter is relevant only for circuit packs of type LOXC, thus the parameter shall be omitted for other packs.
    Type: value set.

Range, if *nvmname* = LOXC

- 0: LOXC in suspended state, initial value after LOXC CP creation
- 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s.

Range, if *nvmname* = LOXC40G2S or LOXC40G3S:

- 0: LOXC in suspended state, initial value after LOXC CP creation
- 15: LOXC active or member of equipment protection group, switch capacity 15 Gbit/s
- 40: LOXC active or member of equipment protection group, switch capacity 40 Gbit/s

Default: 0.

14. **`lostd`** – LO tributary standard
LO tributary standard supported by the LOXC pack. This parameter is relevant only for circuit packs of type LOXC (i.e. *nvmname* = LOXC).
Type: value set.
Range:
    - SDH: LOXC supports SDH LO tributaries (VC3, VC12)
    - SONET: LOXC supports SONET LO tributaries (VT1.5)
    - UNDEFINED (default): no LO tributaries (loswcap = 0).

    LO VC3 tributaries provided with *lostd* = SDH support LO cross-connections and AU3/AU4 conversion cross-connections.
15. **`tmuxcap`** – TransMUX capacity
Switch capacity of the TransMUX pack.
This parameter is relevant only for circuit packs of type TransMUX (that means, *nvmname* = TransMUX), thus the parameter shall be omitted for other packs.
Type: Value set.
Range:
    - 0: TransMUX in suspended state, initial value after TransMUX CP creation.
    - 2G5: TransMUX active or member of equipment protection group, capacity 2.5 Gbit/s

    Default: 0
16. **`istd`** – Interface Standard of the Circuit Pack
Interface standard supported by the circuit pack. This parameter is supported on GE10PL1 and ES64U circuit packs. On the circuit packs, the parameter *istd* defines the interface standard of the virtual concatenation groups.
Option: If the parameter *istd* is not supported, then the parameter is omitted.
Type: Value set.
Range:
    - for GE10PL1: SDH or SONET
    - for ES64U: SONET or UNDEFINED (ES64U not used, *es64ucap* = 0)

Default:

- for GE10PL1: The default is derived from the system parameter *istddflt*.
- for ES64U: UNDEFINED.

Detailed behavior description

Omit Inventory Data parameters if not accessible

If the system cannot access inventory data of a circuit pack, e.g. due to a missing or defect pack or a system-internal communication outage, the following parameters are omitted from the command response:

- **app**
- **clei**
- **eci**
- **name**
- **qual**
- **sln**
- **ssn**.

Example input/output

The following shows the successful execution of the command for a replaceable unit (non CpAID):

```
RTRV-EQPT:UNITE-NE:1-1-usrpnl:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″1-1-usrpnl:name=UPL,qual=-″
   ″1-1-usrpnl:app=XYZ,ssn=S1-3,clei=WMMYABJDAA″
   ″1-1-usrpnl:eci=437442,sln=00NB02160832
;
```

The following shows the successful execution of the command for a circuit pack (CpAID):

```
RTRV-EQPT:UNITE-NE:1-1-#-#-5-cp:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″1-1-#-#-5-cp:eqptpn=DEFAULT,eqptprotpn=DEFAULT″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-EQPT** command.

Related TL1 messages

- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
- "ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
- "DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
- "RTRV-STATE-EQPT (CP): Retrieve equipment state (circuit pack)" (p. 3-2835)

References

GR-199-CORE

RTRV-EQPT (shelf): Retrieve equipment (shelf)

The **RTRV-EQPT** command retrieves the provisioning information associated with the shelf level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of the shelf/subrack.
   Type: AID
   Range: ShelfAID. The keyword **all** is allowed on bay AID and shelf AID level.
   Default: 1-1.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "1-1[:spec_block]"
        ...
        ...
   "1-1[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

1. **`aid`** – Access Identifier
   The AID of the shelf/subrack.
   Type: AID
   Range: ShelfAID. The keyword **`all`** is not allowed on bay AID and shelf AID level.
   Default: 1-1.

Spec block parameters

1. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-UNPROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
2. **`eqptprotpn`** – Protected Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT of the shelf level.
   Type: Alphanumeric string.
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
3. **`shlftp`** – Shelf Type
   The shelf type of the addressed NE is provided.
   Type: Alphanumeric value set.
   Range: LXC40.
   Default: none.
4. **`ssoride`** – SS Bits Overrides
   This parameter determines the value to be inserted into the SS bits of the pointer of every AU-N or STS-N signal in the output direction of every SDH or SONET port in the shelf.

   The terms *override*, *enable* and *disable* are used since:

   - the default value of the SS bits in 00
   - the parameter allows you to diverge from that default.

   Type: Alphanumeric value set.
   Range: ENABLE insert value 00 into SS bits, DISABLE insert value 10 into SS bits.
   Default: DISABLE.

Detailed behavior description

The system retrieves the output parameters as described above.

The shelf type is derived from the inventory data of the subrack.

Example input/output

```
RTRV-EQPT:UNITE-NE:1-1:123456;
```

```
   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ”1-1:eqptpn=DEFAULT,eqptprotpn=DEFAULT,shlftp=LXC40”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to RTRV-EQPT (shelf) command.

Related TL1 messages

- “ENT-EQPT (slot): Enter equipment (slot)” (p. 3-1547)
- “ED-EQPT (shelf): Edit equipment (shelf)” (p. 3-365)
- “DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
- “RTRV-STATE-EQPT (system): Retrieve equipment state (system)” (p. 3-2845)

References

GR-199-CORE

RTRV-EQPT (slot): Retrieve equipment (slot)

The **RTRV-EQPT** command retrieves the provisioning information associated with slot.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a slot.
   Type: AID
   Range:
   - SlotAID
   - UnivSlotAID
   - XCSlotAID
   - CTLSlotAID.

   The keyword **usall** is allowed for Simple_UnivSlotAID.
   The keyword **xcall** is allowed for Simple_XCSlotAID.
   The keyword **ctlall** is allowed for Simple_CTLSlotAID.
   The keyword **all** is allowed for Simple_SlotAID.
   The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
```

```
      ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

The output actlname is an attribute list per slot and nvmname/nvmqual is an optional pair of attributes per slot.

1. **aid** – Access Identifier
   Description: The AID of a slot.
   Type: AID
   Range: SlotAID
   The keyword **usall** is not allowed for Simple_UnivSlotAID.
   The keyword **xcall** is not allowed for Simple_XCSlotAID.
   The keyword **ctlall** is not allowed for Simple_CTLSlotAID.
   The keyword **all** is not allowed for Simple_SlotAID, Simple_UnivSlotAID, Simple_XCSlotAID, Simple_CTLSlotAID.
   The keyword **all** is not allowed on bay/shelf level.

Spec block parameters

1. **actlname** – Acceptable Circuit Pack Name
   This parameter is reported by itself The line is repeated for all groups of acceptable circuit packs in the slot. This list specifies the groups of circuit packs which can be provisioned for that slot. The detailed information about the groups can be requested via RTRV-ACTLCP.
   Type: value set.
   Range: 12 character.
2. **app** – Apparatus Code
   Apparatus Code as stored in the EEPROM on the circuit pack.
   This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
   Type: Alphanumeric string.
   Range: up to 9 characters.
3. **clei** – Common Language Equipment
   Common Language Equipment as stored in the EEPROM on the circuit pack.
   This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
   Type: Value set.
   Range: 10 characters

4. **`eci`** – Equipment Catalog Item
   Equipment Catalog Item as stored in the EEPROM on the circuit pack.
   This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
   Type: value set.
   Range: 6 digits.
   Default: none.
5. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile of type EQUIP-PROT on slot level
   Type: Alphanumeric string
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT.
   Note: *eqptpn* is used on system, shelf and circuit pack level for a profile of type EQUIP-UNPROT.
6. **`name`** – Circuit Pack Name
   Circuit Pack Name as stored in the EEPROM on the circuit pack.
   This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
   Type: value set.
   Range: up to 12 characters.
7. **`nvmname`** – Circuit Pack Name
   This parameter specifies the name of the provisioned circuit pack of the slot. If no circuit pack is provisioned the default value is reported.
   Type: value set.
   Range: 12 character.
   Default: UNDEFINED.
8. **`nvmqual`** – Circuit Pack Qualifier
   This parameter specifies the qualifier of the provisioned circuit pack of the slot. If no circuit pack is provisioned the value UNDEF is reported.
   Type: value set.
   Range: 7 characters.

   The allowed values are:

   - UNDEF (default)
   - EXTD
   - all values as defined in SRD_ConfigurationManagement

   Note: The TL1-representation of the functional qualifier follows the following rules:

   - alphanumeric characters or the special character ’-’ only
   - special characters like ’/’ and ’.’ to be omitted.

9. **`nvmsize`** – NVM Size
   This parameter indicates the size of the NVM installed on the CTL pack. The parameter is reported only for a CTLSlotAID.
   Type: Alphanumeric value set.
   Range:
   - 256MB
   - 512MB
   - 1GB
   - 2GB
   - 4GB
   - UNKNOWN

   The value UNKNOWN is returned, if the NVM size cannot be determined.
10. **`qual`** – Circuit Pack Qualifier
    Circuit Pack Qualifier as stored in the EEPROM on the circuit pack.
    This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
    Type: value set.
    Range: up to 7 characters.
    Note: The TL1-representation of the functional qualifier follows the following rules:
    - alphanumeric characters or the special character '-' only
    - special characters like '/' and '.' to be omitted.
11. **`sln`** – Serial Number
    Serial number as stored in the EEPROM on the circuit pack to identify each circuit pack and indicate the date and place of manufacture.
    This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
    Type: Alphanumeric string.
    Range: 12 characters.
12. **`ssn`** – Series Number
    Series Number as stored in the EEPROM on the circuit pack.
    This parameter should be reported for a SlotAID, if the value of *holdst* is VALID or UNEXPECTED.
    Type: value set.
    Range: up to 6 characters.
    Note: A dash (-) is used instead of a colon (:) for the series number in TL1.

Example input/output

```
RTRV-EQPT:UNITE-NE:1-1-#-#-5:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
```

```
   ”1-1-#-#-5:actlname=OP2G5”
   ”1-1-#-#-5:actlname=OP10”
    ...
   ”1-1-#-#-5:nvmname=OP2G5,nvmqual=13SR4,eqptpn=DEFAULT”
   ”1-1-#-#-5:name=OP2G5,qual=13SR4,app=KFA12”
   ”1-1-#-#-5:clei=SN49CRTBAB,eci=255694”
   ”1-1-#-#-5:sln=00NB01002333,ssn=1-1”
;
```

The following example shows the case when it is not possible to provision slot 5, because *actlname* is empty:

```
RTRV-EQPT:UNITE-NE:1-1-#-#-5:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ”1-1-#-#-5:actlname=\”\””
    ...
   ”1-1-#-#-5:nvmname=OP40,nvmqual=EXTD,eqptpn=DEFAULT”
   ”1-1-#-#-5:name=\”\”,qual=\”\”,app=\”\””
   ”1-1-#-#-5:clei=\”\”,eci=\”\””
   ”1-1-#-#-5:sln=\”\”,ssn=\”\””
;
```

Error responses

Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command in the Error responses section. The error responses listed there also pertain to the **RTRV-EQPT** command.

Related TL1 messages

- “ENT-EQPT (slot): Enter equipment (slot)” (p. 3-1547)
- “ED-EQPT (slot): Edit equipment (slot)” (p. 3-370)
- “DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
- “RTRV-STATE-EQPT (slot): Retrieve equipment state (slot)” (p. 3-2839)
- “RTRV-ACTLCP: Retrieve acceptable circuit pack list” (p. 3-2222).

References

GR-199-CORE

RTRV-EQPT (socket): Retrieve equipment (socket)

The **RTRV-EQPT** command shall retrieve the provisioning information associated with socket.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-EQPT**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_ block]″
   ... ...
   ″aid[:spec_ block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

The output *actlname*/*actlqual* is a list of attribute pairs per socket and *nvmname*/*nvmqual* is an optional pair of attributes per socket.

1. **`aid`** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keywords *all* / *usall* / *scall* shall are not allowed on any level.

Spec block parameters

1. **`actlname`** – Acceptable Module Name
   This parameter is reported by itself together with actlmodqual. The line is repeated for all acceptable modules in the socket. This list specifies the modules which can be provisioned for that socket.
   Type: value set.
   Range: 12 characters.
2. **`actlqual`** – Acceptable Module Qualifier
   This parameter is reported by itself together with actlmodname. The line is repeated for all acceptable modules in the socket. This list specifies the modules which can be provisioned for that socket.
   Type: value set.
   Range: 7 characters.

   Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

   - alphanumeric characters or the special character '-' only
   - special characters like '/' and '.' to be omitted
3. **`app`** – Apparatus Code
   Apparatus Code as stored in the EEPROM. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED. The 9 character Apparatus Code string is represented left aligned and padded on the right with ASCII spaces (20HEX) to a 12 character alphanumeric string.
   Type: Alphanumeric string.
   Range: 12 characters alphanumeric string or empty string (default).
   The default is reported when the Apparatus Code value cannot be derived from the EEPROM content. Note: In TL1 output direction the quotes have to escaped \"\".
4. **`ccode`** – Commercial Code
   Commercial code + suffix as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFPEEPROM Format.
   Type: numeric string.
   Range: up to 12 digits.

5. **`compatibilitybyte`** – Compatibility Byte
   The Compatibility Byte indicates the compatibility of the module with the system. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
   Type: numeric string.
   Range: 0…255.
6. **`clei`** – Common Language Equipment
   Common Language Equipment as stored in the EEPROM on the module.
   This parameter should be reported for a SocketAID, if the value of *holdst* is VALID or UNEXPECTED.
   Type: value set.
   Range: 10 characters.
7. **`connectortype`** – Connector Type
   This parameter indicates the type of connector at the module plugged into the socket. (E.g. 07 for LC) This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
   Type: Hexadecimal number
   Range: 1 byte which is represented as 2 hexadecimal digits.
8. **`datecde`** – Date Code
   Vendor Date Code. This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
   Type: Text string.
   Range: empty string.
9. **`eci`** – Equipment Catalog Item
   Equipment Catalog Item as stored in the EEPROM on the module.
   This parameter should be reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED.
   Type: value set.
   Range: 6 digits.
10. **`eqptpn`** – Equipment ASAP Profile Name
    The name of the current ASAP profile of type EQUIP-PROT on socket level.
    Type: Alphanumeric string.
    Range: up to 24 characters. A hyphen (“-”) is allowed in the name additionally.
    Default: DEFAULT.
11. **`interchangemarker`** – Interchangeability Marker
    This parameter is derived from the EEPROM content on the module and reported for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Test string.
    Range: 6 characters

Default: ″S1-1″
The default is reported when the Interchangeability Marker value is not available in the EEPROM.
Note: A dash (-) is used instead of a colon (:) for the interchangeability marker in TL1.

12. **`name`** – Module Name
Circuit Pack Name as stored in the EEPROM on the module.
This parameter should be reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED.
Type: value set.
Range: up to 12 characters.
13. **`nvmname`** – Provisioned Module Name
This parameter specifies the name of the provisioned module of the socket. If no module is provisioned the default value is reported.
Type: value set.
Range: 12 character.
Default: UNDEFINED.
14. **`nvmqual`** – Provisioned Module Qualifier
This parameter specifies the qualifier of the provisioned module of the slot. If no module is provisioned the default value is reported.
Type: value set.
Range: 7 characters.
Default: UNDEF.
Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:
    - alphanumeric characters or the special character ’-’ only
    - special characters like ’/’ and ’.’ to be omitted.
15. **`physicalidentifier`** – Physical Identifier
This parameter indicates the type of module plugged into the socket. (e.g. 03 for SFP). This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
Type: Hexadecimal number.
Range: 1 byte which is represented as 2 hexadecimal digits.
16. **`prodid`** – Product Line Id
Binary Compatibility + Alcatel-Lucent Unique Number as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string.
Range: empty string ″″ Note: In TL1 output direction the quotes have to escaped \″\″.

17. **`qual`** – Module Qualifier
    Circuit Pack Qualifier as stored in the EEPROM on the module.
    This parameter should be reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED.
    Type: value set.
    Range: up to 7 characters.

    Note: The TL1-representation of the functional qualifier shall follow the rules as defined in SRD_ConfigurationManagement:

    - Alphanumeric characters or the special character '-' only
    - Special characters like '/' and '.' to be omitted.
18. **`sln`** – Serial Number
    Serial number as stored in the EEPROM on the module to identify each module and indicate the date and place of manufacture.
    This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the Small EEPROM Format.
    Type: Alphanumeric string
    Range: 12 characters.
19. **`ssn`** – Series Number
    Series Number as stored in the EEPROM on the module.
    This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the Small EEPROM Format.
    Type: value set.
    Range: up to 6 characters.
    Note: A dash (-) is used instead of a colon (:) for the series number in TL1.
20. **`transceivercode`** – Transceiver Code
    This parameter indicates indicates the type of transceiver at the module plugged into the socket. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM format.
    Type: Hexadecimal number or text string
    Range: 8 bytes where each byte is represented as 2 hexadecimal digits.
21. **`uniquenumber`** – Lucent Unique Number
    Alcatel-Lucent Unique Number as stored in the EEPROM. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED. The Alcatel-Lucent Unique Number is represented right aligned and padded on the left with ASCII spaces (20HEX) to a 12 character numeric string.
    Type: Numeric string.
    Range: 12 characters alphanumeric string or empty string.

22. **`vendorname`** – Vendor Name
This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: empty string ″″
23. **`vendoroui`** – Vendor Company Identifier
This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: empty string ″″
24. **`vendorpartno`** – Vendor Part Number
This parameter is kept for compatibility reasons only and is reported with the fixed value empty string ″″ for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string
Range: empty string ″″
25. **`vendorrevlvl`** – Vendor Revision Level
Vendor Revision Level as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string.
Range: 4 characters.
26. **`vendorsln`** – Vendor Serial Number
Vendor Serial Number + WES Vendor Identifier as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
Type: Text string.
Range: 20 characters.

27. **`wesserialno`** – WES Serial Number
    This parameter is read from the EEPROM on the module and reported for the SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Text string.
    Range: 18 characters.
    Default: empty string "". The default is reported when the WES Serial Number value is not available in the EEPROM.
28. **`wesvendorid`** – WES Vendor ID
    WES Vendor Identifier as stored in the EEPROM on the module. This parameter is reported for a SocketAID, if the value of holdst is VALID or UNEXPECTED and the data in the EEPROM is stored according to the SFP/XFP EEPROM Format.
    Type: Text string.
    Range: 4 characters.

Detailed behavior description

Omit inventory data parameters if not accessible.

If the system cannot access inventory data related to the addressed socket, e.g. due to a missing or defect pack or a system-internal communication outage, the following parameters are omitted from the command response:

- *app*
- *ccode*
- *clei*
- *compatibilitybyte*
- *connectortype*
- *datecde*
- *eci*
- *interchangemarker*
- *name*
- *physicalid*
- *prodid*
- *qual*
- *sln*
- *ssn*
- *transceivercode*
- *uniquenumber*
- *vendorname*
- *vendoroui*
- *vendorpartno*

- *vendorrevlvl*
- *vendorsln*
- *wesserialno*
- *wesvendorid*.

Reporting of inventory data for different module types

Different module types will support a different set of inventory data.

Modules with a Alcatel-Lucent specific specification will support following parameters:

- *clei*
- *eci*
- *name*
- *qual*
- *sln*
- *ssn*.

For SFP and XFP modules the following parameters will be supported:

- *ccode*
- *clei*
- *compatibilitybyte*
- *connectortype*
- *datecde*
- *interchangemarker*
- *name*
- *physicalid*
- *prodid*
- *qual*
- *transceivercode*
- *uniquenumber*
- *vendorname*
- *vendoroui*
- *vendorpartno*
- *vendorrevlvl*
- *vendorsln*
- *wesserialno*
- *wesvendorid*.

Reporting of inventory data based on defined format of EEPROM content

As long as the Inventory Data on the Optical Modules is accessible and readable, the parameters are reported as they are extracted from the EEPROM according to the format definitions. If the data in the EEPROM is not according to the defined format, then for each parameter the defined number of bytes is taken from the defined location in the EEPROM without any check of the extracted data.

Example input/output

The following exampleb shows a normal response for RTRV-EQPT for one socket when a module is plugged in with data stored in Small EEPROM format:

```
RTRV-EQPT:UNITE-NE:1-1-#-#-5-sc2:123456;

   UNITE- NE 00-11-03 08:00:00
M  123456 COMPLD
   ”1- 1-#-#-5-sc2:actlname=OM2G5,actlqual=13LR1”
   ”1- 1-#-#-5-sc2:actlname=OM2G5,actlqual=15LR1”
   ”1- 1-#-#-5-sc2:nvmname=OM2G5,nvmqual=13SR4,eqptpn=DEFAULT”
   ”1- 1-#-#-5:name=OM2G5,qual=15LR1,comcde=109192278000,clei=WM3ILPEDAA,eci=255694,
sln=00NB01002333,ssn=1-1”
;
```

The following example shows a response if a module is not plugged into a socket:

```
RTRV -EQPT:UNITE-NE:1-1-#-#-5-sc2:123456;

   UNITE- NE 00-11-03 08:00:00
M  123456 COMPLD
   ”1- 1-#-#-5-sc2:actlname=OM2G5,actlqual=13LR1”
   ”1- 1-#-#-5-sc2:actlname=OM2G5,actlqual=15LR1”
   ”1- 1-#-#-5-sc2:nvmname=OP2G5,nvmqual=13SR4,eqptpn=DEFAULT”
   ”1- 1-#-#-
      5:name=OM2G5,qual=CL61L1,app=\”\”,ccode=109496869,clei=WM1T37KBAA,compatibilitybyte=128
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-EQPT** command.

RTRV-EQPT (system): Retrieve Equipment (system)

The **RTRV-EQPT** command retrieves the provisioning information on system (NE) level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-EQPT**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system. The usage of the keyword **all** on system level is not allowed.
   Default: system
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-EQPT command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "system:spec_block"
       ...
       ...
   "system:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

1. **`aid`** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system. The usage of the keyword **`all`** on system level is not allowed.
   Default: system

Spec block parameters

1. **`eqptpn`** – Equipment ASAP Profile Name
   The name of the current ASAP profile on the system level
   Type: Alphanumeric string
   Range: up to 24 characters. A hyphen (-) is additionally allowed in the name.
   Default: DEFAULT
2. **`istddflt`** – Interface Standard Default
   Defining the predominantly used interface standard for a port.
   Type: Alphanumeric value set.
   Range:
   - SONET
   - SDH

   Default: SDH
3. **`mondelay`** – Port Auto to Mon Delay
   The switch of the port mode from the value AUTO to the value MON is delayed by this timer. It requires that a valid signal is present for the set timer period prior to the port transitioning to monitored. If during the timer period the signal is interrupted, the time should reset and begin another period when a valid signal is once again presented.
   Type: Numeric string.
   Range:
   - 0 = no timer will be started
   - 1…480 = in steps of 1 representing a switch delay of the port mode in minutes.

   Default: 0
4. **`portclassdflt`** – Port Class Default
   Defines the default port class, which is used when a new port is created and xcsetup=SWISS. When xcsetup=TRADITIONAL this parameter is not evaluated.
   Type: Alphanumeric value set.
   Range:
   - TRADITIONAL
   - EDGE

   Default: TRADITIONAL.
5. **`trbmddflt`** – Tributary Mode Default
   Defining the predominantly used tributary mode for a port.
   Type: Alphanumeric value set.

Range:

- FIXED
- ADAPTIVE

Default: FIXED.

6. **`swmaxcap`** – Maximum Switch Capacity
   This parameter indicates the maximum switch capacity of the system that the user intends to use.
   Type: value set.
   Range:
   - LXC160 (default), 160 GBit/s switch capacity
   - LXC320, 320 GBit/s switch capacity
   - LXC640, 640 GBit/s switch capacity.
7. **`synccrtc`** – System Synchronization Characteristic
   This parameter specifies whether the NE uses the SDH or SONET timing function for synchronization.
   Type: Alphanumeric value set.
   Range:
   - SONET
   - SDH.

   Default: SDH.
8. **`systype`** – System Type
   This parameter specifies the type of NE.
   Type: Alphanumeric value set.
   Range: WaveStar_UNITE.
   Default: WaveStar_UNITE
9. **`upsrportalarming`** – Enh. SA/NSA Detection for UPSR and SNCP
   The parameter specifies whether the Enhanced SA/NSA Detection for UPSR Rings The parameter specifies whether the Enhanced SA/NSA Detection for UPSR Rings is switched on or off.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).

10. **xcsetup** – Cross Connect Setup
    This parameter indicates whether the system supports the set-up of cross-connections via the ONNS application.
    Type: value set.
    Range: SWISS ONNS application is enabled; TRADITIONAL ONNS application is disabled.
    Default: TRADITIONAL.
11. **ASTNNodeID** – ASTN node ID
    This parameter shows the unique identification of the NE as used by the ONNS application. The value will be read from backplane, if DB is empty.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.

Detailed behavior description

The command retrieves the parameters as indicated in the output parameter section from the system.

Example input/output

```
RTRV-EQPT:UNITE-NE:system:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″system:systype=WaveStar_UNITE″
   ″system:istddflt=SDH″
   ″system:trbmddflt=FIXED″
   ″system:synccrtc=SDH″
   ″system:eqptpn=DEFAULT″
;
```

Error responses

Refer to “RTRV-HDR: Retrieve header” (p. 3-2417) command Error Response section. The generic error responses listed there also apply to the **RTRV-EQPT** command.

Related TL1 messages

- “ENT-EQPT (slot): Enter equipment (slot)” (p. 3-1547)
- “ED-EQPT (system): Edit equipment (system)” (p. 3-391)
- “DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
- “RTRV-STATE-EQPT (system): Retrieve equipment state (system)” (p. 3-2845)

References

GR-199-CORE

RTRV-EXT-CONT: Retrieve external control

The **RTRV-EXT-CONT** command instructs an NE to send the control state of MDO ports (external controls). The command can be used to audit the result of an **OPR-EXT-CONT** or an **RLS-EXT-CONT** command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-EXT-CONT**:*tid*:[*aid*]:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDO port entities on which the command operates. If misc_outall is specified the configuration of all MDO’s will be retrieved.
   Type: AID
   Range: Miscellaneous_out_discreteAID
   Default: misc_outall (if the *aid* is omitted).
3. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

The following example shows the successful completion of the **RTRV-EXT-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
   ″aid:,dur,contstate″
   ...
   ″aid:,dur,contstate″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This is the address of the MDO port entity for which the configuration is retrieved or on which an operation is performed.
   Type: AID.
   Range: Miscellaneous_out_discreteAID. misc_outall must not be reported or used.
2. **`dur`** – Duration
   This parameter defines in which mode the device connected to the MDO is expected to be operated. It is a fixed value because this feature will be not supported by the system. This parameter is in for compatibility reason. The system operates only continuous that means if the MDO is operated it is switched on and it is released it is switched off.
   Type: Alphanumeric value set.
   Range: CONTS, continuous duration.
3. **`contstate`** – Control State
   This is the state of the MDO (external control)
   Type: Alphanumeric value set.
   Range:
   - OPER – Operated, that means the MDO is switched on
   - RLS – Released (default), that means the MDO is switched off.

Detailed behavior description

The **`RTRV-EXT-CONT`** command does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of the **`RTRV-EXT-CONT`** command by the NE:

```
RTRV-EXT-CONT:LT-UNITE-123:MISC_OUT3:123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″MISC_OUT3:,CONTS,RLS″
;
```

The following example shows the successful completion of the **`RTRV-EXT-CONT`** command by the NE if the aid is omitted:

```
RTRV-EXT-CONT:LT-UNITE-123::123456;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ″MISC_OUT1:,CONTS,RLS″
   ″MISC_OUT2:,CONTS,OPER″
```

```
   ”MISC_OUT3:,CONTS,OPER”
   ”MISC_OUT4:,CONTS,RLS”
   ”MISC_OUT5:,CONTS,RLS”
   ”MISC_OUT6:,CONTS,OPER”
   ”MISC_OUT7:,CONTS,RLS”
   ”MISC_OUT8:,CONTS,RLS”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-EXT-CONT** command.

Related TL1 messages

- “OPR-EXT-CONT: Operate external control” (p. 3-2032)
- “RLS-EXT-CONT: Release external control” (p. 3-2190).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

RTRV-FECOM: Retrieve DCC related parameters

The **RTRV-FECOM** command can be initiated by a user to retrieve parameters of Data Communication Channels (DCC) and its attachment to specific OSI nodes.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-FECOM***:tid::ctag::[port],[dcctype],[networktype];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **port** – DCC port
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID.
   Range: SPortAID.
   Default: If this field is omitted, the report will include the DCC parameters of all transmission ports belonging to the NE.

4. **dcctype**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC parameters of the *port* is reported by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.
   Default: If this parameter is omitted, LINE and SECTION DCC parameters will be considered for the output.
5. **networktype** – Network Type
   This parameter specifies whether the MCN or SCN related DCC parameters of the port is reported by the command.
   Type: Alphanumeric value set.
   Range: MCN, SCN.
   Default: If this parameter is omitted, MCN and SCN related parameters will be considered for the output.

Output format

If the RTRV-FECOM is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″port,dcctype:spec_block″
        ...
        ...
        ...
   ″port,dcctype:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`port`** – DCC port
   This parameter indicates the port AID of a circuit pack which is used together with *dcctype* to identify the DCC channel.
   Type: AID.
   Range: SPortAID.
   Default: If this field is omitted, the report will include the DCC parameters of all transmission ports belonging to the NE.
2. **`dcctype`**
   This parameter specifies whether the LINE/MS or SECTION/RS DCC parameters of the *port* is reported by the command.
   Type: Alphanumeric value set.
   Range: LINE, SECTION.
   Default: If this parameter is omitted, LINE and SECTION DCC parameters will be considered for the output.

Spec block parameters

1. **`dccprottp`**
   This parameter reports the protection type of the DCC channel. 0X1 means the port is not involved in a protection group or DCC protection switching is disabled. 1+1 means DCC protection switching is enabled: In these case only the active worker line is shown.
   Type: Alphanumeric value set.
   Range: 0X1 (default), 1+1.
2. **`dccstat`**
   This parameter indicates whether MCN communication over the DCC channel is enabled or if it is used in a Transparent DCC Cross Connection.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE (default), TRANSPARENT.
   Note: Output responses with dccstat=DISABLE will only sent when either *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE or *enni_dccstat* = ENABLE, see also Detailed behavior description below.
3. **`lapdmode`**
   This parameter indicates the type of LAPD service. Modes are AITS and UITS. Starting with R2.0 this parameter is also used to determine the service for the LinkID protocol.
   If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* is set to ENABLE, or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set..
   Range:
   - UITS (default)
   - AITS.

4. **lapdrl**
   This parameter indicates the LAPD role. This parameter is also used to determine the service for the LinkID protocol.
   If the DCC channel is involved in a transparent DCC connection or if either *scn_dccstat* or *uni_dccstat* is set to ENABLE or *enni_dccstat* is set to ENABLE, this parameter has no meaning and will not be reported.
   Type: Alphanumeric value set.
   Range:
   - USER-SIDE (default)
   - NETWORK-SIDE.
5. **osinode**
   Assignment of a DCC channel to an MCN node in the NE. The parameter is omitted when *dccstat* = ENABLE.
   An NE can have up to 8 instances of the OSI stack. Node-1 is default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00). Nodes 2-8 can be created and configured, see command **ENT-ULSDCC-L3**.
   Type: AID.
   Range: osinode{1-8}
6. **scn_dccstat**
   This parameter indicates whether the DCC channel is used for communication over the SCN or is involved in a transparent DCC cross-connection.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT.
   Note: Output responses with *scn_dccstat* = DISABLE will only sent when either *dccstat* = ENABLE or *uni_dccstat* = ENABLE, or *enni_dccstat* = ENABLE, see also Detailed behavior description below.
7. **uni_dccstat**
   This parameter indicates whether the DCC channel is used for UNI communication.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT
   Note: Output responses with *uni_dccstat* = DISABLE will only sent when either *dccstat* = ENABLE or *scn_dccstat* = ENABLE, or *enni_dccstat* = ENABLE, see also Detailed behavior description below.
8. **enni_dccstat**
   This parameter indicates whether the DCC channel is used for E-NNI communication.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE, TRANSPARENT
   Note: Output responses with *enni_dccstat* = DISABLE will only be sent when either *dccstat* = ENABLE or *scn_dccstat* = ENABLE or *uni_dccstat* = ENABLE, see also Detailed Behavior Description below.

Detailed behavior description

The command retrieves the system parameter as described in the Output Parameter Section.

Output responses will only be delivered for DCC channels with

- *dccstat!* = DISABLE or

- *scn_dccstat!* = DISABLE or

– *uni_dccstat*!=DISABLE or

– *enni_dccstat*!=DISABLE.

The command shall return an empty response, if there is no data matching the criteria given by the input parameters of the command.

The command processing uses the input parameters as filter criteria to search for data to be retrieved. If no data is found, an empty response is returned. Error messages are only returned for general syntax and range errors.

The command is not able to retrieve DCC characteristics parameters from the protecting port of a 1+1 MSP group, if DCC protection switching is enabled.

Example input/output

The following example shows the result of RTRV-FECOM:

```
RTRV-FECOM:NE-UNITE::123456::1-1-#-#-5-all,line;
   NE-UNITE 00-12-24 11:00:00
M  123456 COMPLD
   ″1-1-#-#-5-1,LINE:dccstat=ENABLE,lapdmode=UITS,osinode=osinode1,
dccprottp=0X1,″
;
```

The following example shows the result of RTRV-FECOM for a port which is used by the SCN:

```
RTRV-FECOM:NE-UNITE::123456::1-1-#-#-5-all,line;
   NE-UNITE 00-12-24 11:00:00
M  123456 COMPLD
   ″1-1-#-#-5-1,LINE:dccstat=DISABLE,dccprottp=0X1,scn_dccstat=ENABLE, uni_
      dccstat=DISABLE, enni_dccstate=DISABLE″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-FECOM** command.

Related TL1 messages

- "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
"RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection" (p. 3-2329)

RTRV-FECOM-LAN: Retrieve LAN channel related parameters

The **RTRV-FECOM-LAN** command can be initiated by users to retrieve the provisioned state of the LAN ports (LAN1, LAN2, and LAN3).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-FECOM-LAN**:*tid:[aid]:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   Default: If this field is omitted, the report will display the status of all lan ports belonging to the *tid*.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.

Output format

If the **RTRV-FECOM-LAN** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:spec_block"
        ...
   "aid:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Output parameters

Command parameters

1. **aid** – Target Identifier
   This parameter indicates the LAN port AID. The value *lanall* is not allowed.
   Type: AID.
   Range: LanPort AID.

Spec block parameters

1. **ip_ad** – IP Address
   This parameter indicates the assigned IP address for the LAN interface. The IP-address includes both, a router address and a HOST address. The HOST address that will be used for FTP depends on the LAN interface where the related TCP connection request originates.
   The CIT is expected to use the HOST address of LAN1. The *Navis*® Optical EMS is expected to use the HOST address of LAN2 (or LAN3 in later releases).
   Type: 32-bit IPv4 address: Four dot separated numeric strings OR empty string The empty string value is used as indication for an invalid/undefined IP address.
   Range: For first numeric string: 0 .. 223; for second to fourth numeric string: 0 .. 255.
   The values 0.0.0.0 and 1.1.1.1 are excluded.
   Default: For LAN1, LAN2 and LAN3: empty string.
2. **ip_lanstat** – IP-LAN Status
   This parameter indicates whether IP communication over the external LAN channel is enabled or disabled. Note: Enabling has no effect if *lanstat* is disabled for that interface.
   Type: Alphanumeric value set.
   Range: ENABLE or DISABLE for LAN1, LAN2, LAN3
   Default: DISABLE for LAN1, LAN2, LAN3.
3. **ip_snm** – IP Subnet Mask
   This parameter indicates the assigned IP subnet mask (32-bit IPv4) for the LAN interface. The full IP-address is needed as host-address. Each LAN needs its own IP-address and IP-mask, indicating that all IP-addresses that match the prefix (Unmasked part) are grouped as the subnet-ID for this LAN-subnet.
   If the LAN interface is used for UNI, then the parameter ip_snm is used to define a subnet for multiple UNI-Cs which belong to the same customer.
   Type: 32-bit IPv4 address: Four dot separated numeric strings or empty string. The empty string value is used as indication for an invalid/undefined IP subnet mask.
   Range: For each numeric string: 0 .. 255.
   Default: For LAN1, LAN2, and LAN3: empty string.

4. **`isis_stat`** – (I) IS-IS Status
   This parameter indicates whether (integrated) IS-IS is enabled or disabled on the LAN port specified by the input *aid*.
   Note: Enabling has no effect if *lanstat* is disabled for that interface.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE.
5. **`lanrpri`** – LAN Designated Router Priority
   This parameter sets the relative priority for the node in the election for designed router on this LAN. There is a separate election for each separate LAN to which the node is connected. A higher value indicates a higher probability to be selected as designated router. A designated router is selected per area (between the nodes on that LAN with the same area address) and for the Level-2 nodes on the LAN.
   Note: When *osinode* = scnnode modification of this parameter has no effect.
   Note: When *osinode* = uninode this parameter has no effect.
   Note: When *osinode* = enninode this parameter has no effect.
   Type: numeric string.
   Range: 1 – 127.
   Default: 64.
6. **`lanstat`** – LAN Status
   This parameter indicates whether physical communication over the external channel is enabled or disabled.
   Note: Disabling of an interface also affects usage of protocols over this interface, see *lan_stat* and *osi_stat*.
   Type: Alphanumeric value set.
   Range: ENABLE for lan1; ENABLE, DISABLE for lan2, lan3:
   Default: ENABLE for lan1 and lan2; DISABLE for lan3.
7. **`lantlss`** – LAN TARP Storm Suppression
   This parameter indicates the enhanced TARP propagation rules that suppress TARP LAN storms.
   Note: When *osinode* = uninode this parameter has no effect.
   Note: When *osinode* = enninode this parameter has no effect.
   Type: Alphanumeric value set.
   Range:
   - ENABLE (default)
   - DISABLE.
8. **`osinode`**
   Assignment of LAN port to an OSI-node, the SCN node, the UNI node, or the E-NNI node in the NE. A UNITE NE can have up to 8 instances of the OSI stack, has one SCN node, one UNI node, and one E-NNI node.
   Node-1 is the default and has a NSAP. Nodes 2-8 have a NET and no SEL (NSAP = NET + SEL, for NET: SEL=00).
   Node 1 is always present. The SCN node is created when the ASTN IP address is configured, the UNI node is created when the UNI IP address is configured, the SC node (ENNI node) is created when the SC IP address and the SC ID is configured.

Node 2-8 can be created and configured, see command **ENT-ULSDCC-L3**.
Type: AID.
Range:
For LAN1: osinode1 (fixed value).
For LAN2 and LAN3: osinode{1-8} | scnnode | uninode | enninode.
Note: If this parameter is set to scnnode or uninode or enninode, then *lanstat* and *ip_lanstat* must be ENABLED.
Default: osinode1.

9. **osi_lanstat** – OSI LAN Status
This parameter indicates whether OSI communication over the external channel is enabled or disabled.
Type: Alphanumeric value set.
Range: ENABLE for LAN1, LAN2, LAN3

Detailed behavior description

The command retrieves system parameter as described in the Output parameter section.

Example input/output

The following example shows the result of RTRV-FECOM-LAN:

```
RTRV-FECOM-LAN:NE-UNITE:1-1-lan1:123456;
   NE-UNITE 00-12-24 11:01:00
M  123456 COMPLD
   ″1-1-lan1:ip_lanstat=ENABLE,ip_ad=135.248.206.72,ip_snm=255.255.0.0,
lanstat=ENABLE,lanrpri=64,lantlss=ENABLE,osinode=osinode1,osi_lanstat=ENABLE″
;
```

The following example shows the result of RTRV-FECOM-LAN:

```
RTRV-FECOM-LAN:NE-UNITE:1-1-lan1:123456;
   NE-UNITE 00-12-24 11:01:00
M  123456 COMPLD
   ″1-1-lan2:ip_lanstat=ENABLE,ip_ad=135.248.206.72,ip_snm=255.255.0.0,
lanstat=ENABLE,lanrpri=64,lantlss=ENABLE,osinode=enninode,osi_lanstat=ENABLE″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-FECOM-LAN** command.

Related TL1 messages

- “ENT-FECOM-LAN: Enter LAN related parameters” (p. 3-1581)

RTRV-HDR: Retrieve header

Description

The **RTRV-HDR** command can be initiated by a user to request that the NE returns a normal completion response.

Parameters

In TL1 terminology, a parameter block always follows a colon ’:’. Any trailing colons at the end of a command line may be omitted if they do not contain any parameters. A parameter block can be either a block of position-defined parameters or it can be a spec_block (a block of name-defined parameters). In a block of position-defined parameters, the parameter values are supplied at predefined positions within the parameter block. Two parameter values are separated by a comma. If the default value for a parameter is to be used, a null value is placed in the command line and the position of the parameter is maintained by keeping the comma. When default values are used for parameters at the end of the parameter block, it is not necessary to include all the trailing commas. In all the TL1 command pages, for position-defined parameter blocks, simplified notation such as: [d],[e],[f],[g] is used to represent the following format: [d][,[e][,[f][,g]]]. The following notation is also equivalent: [d],[e],[f][,g]. In a specification block (spec_block), every parameter name is followed by an equal sign and the parameter value. The command line does not have to use all of the spec_block parameters. Also, spec_block parameters can appear in any order.

NULL parameter value

The word NULL is used for some of the parameter values in TL1 messages. For both input and output parameters, NULL means that the parameter is omitted.

Quoted text strings

Some string-valued parameters, like NENAME, allow any printable character within the double-quoted string (″″). On output (RTRV and **REPT DBCHG**), the string is surrounded by escaped quotes \″\″. Those parameters can contain any printable characters, including the TL1 parse characters (; : , ″). Spaces and upper/lowercase are also preserved. Using two double-quotes ″″ will set the parameter to an empty string. But according to GR-831, within a quoted string, a double-quote character is escaped as \″ and the backslash is escaped as \\\. Within a backslash-quoted string (\″\″), the double-quote character is escaped as ″″ and the backslash is escaped as \\\. Another caveat is that while that parameter can contain any printable character, in networks using a Telcordia OS, the TL1 parse characters (; : , ″) should not be used. Particularly, the semicolon (;), less-than (<), and greater-than (>) characters must not be used in the string.

Other notational conventions

The following conventions are used in the description of RTRV-HDR itself to visualize non-printable characters (in all other commands those characters are indicated by a blank).

<cr>

A carriage return in ASCII

<lf>

A line feed in ASCII

<>

The brackets are used to enclose one or more information items, e.g., <date>, <time>, <source identifier>, etc. The brackets are not transmitted.

^

The character ^ indicates a blank that must appear in the message.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): Any

Priority: 1

Abortable: Yes

Input format

Syntax: `RTRV-HDR`*:tid::ctag*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   This is the currently active name of the NE to which the command is addressed. The *tid* is a case insensitive character string.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed
   Range: 1 to 20 characters
   Default: UNITE-NE
2. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string.
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows '0' as valid *ctag*, which is an extension to GR-831.

Output format

Normal system response

After receiving the **RTRV-HDR** command, the following normal completion response is returned:

```
IP^ctag<cr><lf>
<cr><lf><lf><lf>
^^^sid^date^time<cr><lf>
M^^ctag^COMPLD<cr><lf>
;
```

The format definition shows only the minimum of <CR> characters needed for the response. The response may contain additional <CR> characters at the beginning or end of lines. These <CR> characters have no effect on the visible formatting of the response message.

RL Acknowledgment

If a command cannot be executed at the present time, the following RL (Repeat Later) Acknowledgment is returned:

```
RL^ctag<cr><lf>
<cr><lf><lf>
```

Continuation responses and IP acknowledgments

For commands that generate output, if the output response is less than or equal to 4096 bytes, the response will be terminated by the semicolon (;) character. If the output response is greater than 4096 bytes, the response lines will be partitioned into multiple responses. The continuation response will have another set of header lines with the same ctag, along with additional output response lines. Each response will be terminated with the greater than (>) character, except the last one which will be terminated by the semicolon (;) character.

Output parameters

1. **sid** – Source Identifier
   This is the currently active name of the NE to which responses to the command. See definition of *tid* input parameter in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **date** – Date
   Date when output message is generated. This has the format YY-MM-DD (year-month-day).
   Type: value set.

Range:

- for YY: 00 to 99
- for MM: 01 to 12
- for DD: 01 to 31

2-digit years
TL1 messages use YY-MM-DD for the format of a date, meaning that the year only has two digits. This is consistent with Telcordia requirements. Internally, however, the system does use a 4-digit year. When the year appears in a response, the internal 4-digit year is truncated to two digits. When the year appears as an input parameter, for example in the **ED-DAT** command, years in the range 90-99 will represent 1990-1999, and years in the range 00-37 will represent 2000-2037.
Default: 00-01-01

3. **time** – Time
   Time when output message is generated. This has the format HH:MM:SS (hours:minutes:seconds).
   Type: Value set.
   Range:
   - for HH: 00 to 23
   - for MM: 00 to 59
   - for SS: 00 to 59

   Default: 00:00:00
4. **M** – Message
   This parameter indicates that the output message is generated in response to a manual command.
   Type: Value set.
   Range: M
5. **ctag** – Correlation Tag
   See definition of *ctag* input parameter in "RTRV-HDR: Retrieve header" (p. 3-2417).
6. **IP** – In Progress
   IP is sent in response to all commands with the exception of commands that result in error code output.
   Type: value set.
   Range: IP

7. **RL** – Repeat Later
   The requested command cannot be executed due to unavailable system resources. The command may be entered again later.
   Type: value set.
   Range: RL.
8. **COMPLD** – Completion Code
   This parameter indicates that the command has been completed.
   COMPLD for successful command execution For a command that is decomposed into a set of atomic commands a COMPLD completion code means that the command was successful for all atomic commands (e.g. AID ranges, group alias).
   COMPLD for no changes If a parameter has a current value, and a provisioning command attempts to reprovision the parameter with a value that is the same as its current value, the command will return a COMPLD completion code.
   Type: value set.
   Range: COMPLD.

Detailed behavior description

The **RTRV-HDR** command returns a normal completion response with *sid*, *date* and *time* output parameter as described above.

Example input/output

```
RTRV-HDR:LT-UNITE::123456;
IP 123456
<
   LT-UNITE 00-11-03 08:00:00
M  123456 COMPLD
;
```

Error responses

The following error conditions are not unique to the **RTRV-HDR** command but may apply to any TL1 command. Other commands will use the defined command specific error codes with additional error texts.

If there is a specific error behavior defined for another command, e.g. other error code or the suppression of error responses in some cases, then the specific error behavior, which is defined together with the command, shall overrule the generic behavior defined in this chapter.

There is no implied priority order of error checking suggested by this list of error responses.

In general the *sid* value in the error responses is that of the addressed NE. This is true even if a TL1-GNE sends an error message on behalf of an RNE. Some exceptions are noted for some error codes below (e.g. IITA). If the TL1-GNE forwards a command to a remote NE and a communication failure occurs before the remote NE is able to respond, it is understood that no response may be returned.

When an error response is returned, there may not be an in-progress (IP) acknowledgment in the response.

Note, that validation rules for the generic error conditions described below are not repeated for each specific command in the Error response section.

General format

The general output format is shown below, where <error code> is a 4-character code and <error text> is a text string.

```
<cr><lf><lf>
   sid date time<cr><lf>
M  ctag^DENY<cr><lf>
   <error code><cr><lf>
   /*<error text[, additional command specific error text]>*/<cr><lf>
;
```

Validation rule: Input failure (ICNV)

If there was an input failure because the command cannot be processed, the following error response is returned:

Error code: `ICNV – Input, Command Not Valid`

```
   sid date time
M  ctag DENY
   ICNV
   /* Input, Command Not Valid */
;
```

Validation rule: Input failure for omitted TID

If a *tid* value is required but no *tid* value is included, the following error response is returned, using the *sid* of the system that detected the error:

Error code: `ICNV – Input, Command Not Valid, missing TID`

```
   sid date time<cr><lf>
M  ctag DENY
   ICNV
   /*Input, Command Not Valid, missing TID*/
;
```

Validation rule: Command not recognized (SCNF)

If the command is not recognized or command parameters are not valid, the following error response is returned:

Error code:
`SCNF – Status, Command Not Found, command not recognized or command parameters not valid`

```
   sid date time
M  ctag DENY
   SCNF
   /* Status, Command Not Found, command not recognized
or command parameters not valid */
;
```

Validation rule: Syntax or Punctuation Error (IISP)

If the command cannot be parsed due to a syntax or punctuation error, the following error response is returned. If the command cannot be parsed for at least 3 colons, and no more than the maximum number of colons allowed before the TL1 end-of-message semicolon, the following error response is returned:

Error code: `IISP – Input, Invalid Syntax or Punctuation`

```
   sid date time
M  ctag DENY
   IISP
   /* Input, Invalid Syntax or Punctuation */
;
```

Validation rule: Command too long

If the command is longer than 4096 characters, the following error response is returned:

Error code: `IISP – Input, Invalid Syntax or Punctuation, command too long`

Validation rule: Parameter Input Failure (INUP)

If there is an input failure due to the parameter types, the following error response is returned:

Error code: `INUP – Input, Non-null Unimplemented Parameter`

```
   sid date time<cr><lf>
M  ctag DENY
   INUP
   /* Input, Non-null Unimplemented Parameter */
;
```

Validation rule: Parameter Missing (IPMS)

If an input parameter value is missing, the following error response is returned.

Error code: `IPMS – Input, Parameter MiSsing,<parameter_name> is missing`

```
   sid date time
M  ctag DENY
   IPMS
   /* Input, Parameter MiSsing */
;
```

Validation rule: Any Input Parameter not valid (IDNV)

If any one or more of the input parameter values are not valid, the following error response is returned:

Error code: `IDNV – Input, Data Not Valid`

```
   sid date time
M  ctag DENY
   IDNV
   /* Input, Data Not Valid */
;
```

Validation rule: Input Parameter not valid (IDNV)

If an input parameter value is not valid, the following error response is returned. This is the preferred error response.

Error code: `IDNV – Input, Data Not Valid, invalid <parameter_name>`

```
   sid date time<cr><lf>
M  ctag DENY
   IDNV
   /* Input, Data Not Valid, invalid <parameter name> */
;
```

Validation rule: Invalid Entity (IENE)

If there is an input failure due to an invalid entity, the following error response is returned:

Error code: `IENE – Input, Entity Not Exists`

```
   sid date time
M  ctag DENY
   IENE
   /* Input, Entity Not Exists  */
;
```

Validation rule: Invalid Command Qualifier (IPNC)

If there is an input failure due to an invalid command qualifier, the following error response is returned:

Error code. `IPNC – Input, Parameter Not Consistent`

```
   sid date time
M  ctag DENY
   IPNC
   /* Input, Parameter Not Consistent */
;
```

Validation rule: Invalid Instance (SDNC)

If there is an input failure due to an invalid instance of an entity, the following error response is returned:

Error code: `SDNC – Status, Data Not Consistent, invalid instance of entity`

```
   sid date time
M  ctag DENY
   SDNC
   /* Status, Data Not Consistent,invalid instance of entity */
;
```

Validation rule: Incorrect TID (IITA)

If the command has a syntactically incorrect *tid*, the following error response is returned, using the *sid* of the system that detected the error:

Error code: `IITA – Input, Invalid TArget identifier, incorrect syntax`

```
   sid date time
M  ctag DENY
   IITA
   /* Input, Invalid TArget identifier,incorrect syntax */
;
```

Validation rule: Unknown TID (IITA)

If a command contains an unknown tid value (a *tid* value that does not exist in the network), the following error response is returned, using the *sid* of the system that detected the error:

Error code: `IITA – Input, Invalid TID, Unknown TID`

```
   sid date time
M  ctag DENY
   IITA
   /* Input, Invalid TID, Unknown TID */
;
```

Validation rule: Not reachable TID (IITA)

If a command has a valid *tid* value but the addressed remote NE is temporarily unable to communicate with the TL1-GNE, the following error response is returned, using the *sid* of the addressed system:

Error code: `IITA – Input, Invalid TID, Communications Failure`

```
   sid date time
M  ctag DENY
   IITA
   /* Input, Invalid TID,Communications Failure */
;
```

Validation rule: Too much AID parameters (ICNV)

If the command line has more aid parameters in the *aid* block than are allowed for that command, the following error response is returned:

Error code: `ICNV – Input, Command Not Valid, extra AID`

```
   sid date time
M  ctag DENY
   ICNV
   /* Input, Command Not Valid, extra AID */
;
```

Validation rule: Missing AID (ICNV)

If the command line has fewer aid parameters in the *aid* block than are required for the command, the following error response is returned:

Error code: `ICNV – Input, Command Not Valid, required AID missing`

```
   sid date time
M  ctag DENY
   ICNV
   /* Input, Command Not Valid, required AID missing */
;
```

Validation rule: Incorrect AID Syntax (IIAC)

For commands that have aid input parameters in the *aid* block, if the aid is syntactically incorrect, the following error response is returned:

Error code: `IIAC – Input, Invalid ACcess identifier, incorrect syntax`

```
   sid date time
M  ctag DENY
   IIAC
   /* Input, Invalid ACcess identifier, incorrect syntax */
;
```

Validation rule: Unknown AID (IIAC)

For commands that have aid input parameters in the *aid* block, if the aid is valid syntactically but has an unknown value, the following error response is returned:

Error code: `IIAC – Input, Invalid ACcess identifier, unknown AID`

```
   sid date time
M  ctag DENY
   IIAC
   /* Input, Invalid ACcess identifier, unknown AID */
;
```

Validation rule: Missing CTAG (ICNV)

If the *ctag* value is missing, the following error response is returned. Note that the *ctag* value is 0 in the error response:

Error code: `ICNV – Input, Command Not Valid,missing CTAG`

```
   sid date time
M  0 DENY
   ICNV
   /* Input, Command Not Valid,missing CTAG */
;
```

Validation rule: Invalid CTAG (IICT)

If the *ctag* value is non-NULL but invalid, the following error response is returned. Note that the same invalid *ctag* value is used for the *ctag* field in the error response:

Error code: `IICT – Input, Invalid Correlation Tag (CTAG), incorrect syntax`

```
   sid date time
M  ctag DENY
   IICT
   /* Input, Invalid Correlation Tag (CTAG), incorrect syntax */
;
```

Validation rule: Input Parameter Range Incorrect (IDRG)

If an input parameter accepts a range but the given range is incorrect, the following error response is returned:

Error code: `IDRG – Input, Data Range Error`

```
   sid date time
M  ctag DENY
   IDRG
   /* Input, Data, Range Error */
;
```

Validation rule: Invalid File or Directory Name (SDNC)

If the command failed due to an invalid file or directory name, the following error response is returned:

Error code: `SDNC – Status, Data Not Consistent, invalid file or directory`

```
   sid date time
M  ctag DENY
   SDNC
   /* Status, Data Not Consistent, invalid file or directory */
;
```

Validation rule: Invalid Input Parameter Value Combination (IDNC)

If a command has more than one input parameter, but the combination of values is not valid, the following error response is returned:

Error code `IDNC – Input, Data Not Consistent`

```
   sid date time
M  ctag DENY
   IDNC
   /* Input, Data Not Consistent */
;
```

Validation rule: Missing Equipage (ENEQ)

If the command failed due to equipage that is missing, the following error response is returned:

Error code: `ENEQ – Equipage, Not EQuipped`

```
   sid date time
M  ctag DENY
   ENEQ
   /* Equipage, Not EQuipped */
;
```

Validation rule: Insufficient File Space (ENEQ)

If the command failed due to insufficient file space, the following error response is returned:

Error code: `ENEQ – Equipage, Not EQuipped, insufficient file space`

```
   sid date time
M  ctag DENY
   ENEQ
   /* Equipage, Not EQuipped, insufficient file space */
;
```

Validation rule: Incorrect Equipage (EATN)

If the command failed due to incorrect equipage, the following error response is returned:

Error code: `EATN – Equipage, not valid for Access Type, incorrect equipage`

```
   sid date time
M  ctag DENY
   EATN
   /* Equipage, not valid for Access Type, incorrect equipage */
;
```

Validation rule: TID Mismatch in retrieved File (EATN)

If the command failed due to a TID mismatch in a retrieved file, the following error response is returned:

Error code:

`EATN – Equipage, Not Valid for Access Type, retrieved file TID mismatch`

```
   sid date time
M  ctag DENY
   EATN
   /* Equipage, Not Valid for Access Type, retrieved file TID mismatch    */
;
```

Validation rule: Product Mismatch (SRAC)

If the command failed due to a product mismatch in the retrieved file, the following error response is returned:

Error code:

`SRAC – Status, Requested Access Configuration is invalid, retrieved file NE mismatch`

```
   sid date time
M  ctag DENY
   SRAC
   /* Status, Requested Access Configuration is invalid, retrieved    file NE
      mismatch */
;
```

Validation rule: Failed Target Hardware (SSTP)

If the command failed due to failed target hardware, the following error response is returned:

Error code: `SSTP – Status, Stopped, failed target hardware`

```
   sid date time
M  ctag DENY
   SSTP
   /* Status, Stopped, failed target hardware */
```

```
;
```

Validation rule: Command denied based on Mode, State or Status (SNVS)

If the command is denied based on mode, state, or status, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State`

```
   sid date time
M  ctag DENY
   SNVS
   /* Status, Not in Valid State */
;
```

Validation rule: Command denied based on Maintenance Condition (SNVS)

If the command is denied based on the maintenance condition, the following error response is returned:

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

```
   sid date time
M  ctag DENY
   SNVS
   /* Status, Not in Valid State, not in Maintenance Condition */
;
```

Validation rule: Entity already exists (IEAE)

If an entity already exists, the following error response is returned:

Error code: `IEAE – Input, Entity Already Exists`

```
   sid date time
M  ctag DENY
   IEAE
   /* Input, Entity Already Exists */
;
```

Validation rule: Session is not initiated (PLNA)

If a user issues a command before a session is initiated with ACT-USER, the following error response is returned:

Error code: `PLNA – Privilege, Login Not Active`

```
   sid date time
M  ctag DENY
   PLNA
   /* Privilege, Login Not Active */
;
```

Validation rule: Command requires higher privilege (PICC)

If a command requires a privilege higher than that of the user issuing the command, or for any other security-related problems, the following error response is returned:

Error code: `PICC – Privilege, Illegal Command Code`

```
   sid date time
M  ctag DENY
   PICC
   /* Privilege, Illegal Command Code */
;
```

Validation rule: System Initializing not completed (SNVS)

If a command is not allowed because the system has not completed initializing, the following error response is returned:

Error code: `SNVS – Status, Not in Valid State, system initializing`

```
   sid date time
M  ctag DENY
   SNVS
   /* Status, Not in Valid State, system initializing */
;
```

Validation rule: System Limit exceeded (SARB)

If the system is experiencing temporary exhaustion of allocated resources, the following error response is returned:

Error code: `SARB – Status, All Resources Busy, system limit exceeded`

```
   sid date time
M  ctag DENY
   SARB
   /* Status, All Resources Busy, system limit exceeded */
;
```

Validation rule: System Busy with SW Generic Installation (SARB)

If the command cannot be executed because the system is prepared for a SW Generic installation with fall-back option, the following error response is returned:

Error code: `SARB – Status, All Resources Busy`

```
   sid date time
M  ctag DENY
   SARB
   /* Status, All Resources Busy */
;
```

Validation rule: Internal Limit exceeded (SSRE)

If a command attempts to exceed an internal limit, the following error response is returned:

Error code:

`SSRE – Status, System Resources Exceeded, allowed limit exceeded`

```
   sid date time
M  ctag DENY
   SSRE
   /* Status, System Resources Exceeded, allowed limit exceeded */
;
```

Validation rule: Operation Failed (SROF)

If the command cannot be completed due to system difficulties other than hardware or equipment, the following error response is returned:

Error code: `SROF – Status, Requested Operation Failed`

```
   sid date time
M  ctag DENY
   SROF
   /* Status, Requested Operation Failed */
;
```

Validation rule: Processing Failed (SROF)

If the command cannot be completed due to internal system processing problems, the following error response is returned:

Error code: `SROF – Status, Requested Operation Failed, processing failure`

```
   sid date time
M  ctag DENY
   SROF
   /* Status, Requested Operation Failed, processing failure */
;
```

Validation rule: Hardware Failed (SWFA)

If the command failed because the control hardware is failed, missing, or initializing, the following error response is returned:

Error code:

`SWFA – Status, Working unit Failed, control hardware failed, missing or initializing`

```
   sid date time
M  ctag DENY
   SWFA
   /* Status, Working unit Failed, control hardware failed, missing or
   initializing */
```

```
;
```

Validation rule: Communication Failure (SROF)

If the command failed due to an external communications failure, the following error response is returned:

Error code:
`SROF – Status, Requested Operation Failed, external communications failure`

```
   sid date time
M  ctag DENY
   SROF
   /* Status, Requested Operation Failed, external communications
   failure */
;
```

Validation rule: Generic not available (ENSG)

If the command failed because the generic is not available, the following error response is returned:

Error code: `ENSG – Equipage, Not Software Generic, generic not available`

```
   sid date time
M  ctag DENY
   ENSG
   /* Equipage, Not Software Generic, generic not available */
;
```

Validation rule: Command Aborted (SABT)

If a user or outside agent has aborted the command, the following error response is returned:

Error code: `SABT – Status, Aborted`

```
   sid date time
M  ctag DENY
   SABT
   /* Status, Aborted */
;
```

Partial successful completion of a set of atomic commands

For provisioning commands (non-RTRV commands) that operate on a range of entities (e.g. AID range, or via an group alias) range as input, if the command succeeded on only a non-empty subset of the AIDs or another set of atomic commands, the following error response is returned for the AIDs or atomic commands that were not completed successfully. Note that the completion code is PRTL, not DENY.

```
   sid date time
```

```
M  ctag PRTL
   ″aid″
   <error code>
   /* <error text> */
   ″aid″
   <error code>
   /* <error text> */
   ...
   ″aid″
   <error code>
   /* <error text> */
;
```

The error codes and error text fields are as defined in the other error responses above. If the command failed on all the AIDs, the completion code will be DENY, and there will only be a single error code and error text in the response.

Partial completion on RTRV commands

For RTRV commands that take an AID range as input, if the command fails on a non-empty subset of the AIDs, and the failure is not related to filtering, the entire command will fail and one error response is returned. If the error is related to filtering, the command completes successfully but only the data that matches is returned.

If there is no data matching the filter criteria, the NE just responds with an empty reply.

For example, RTRV-OC48 with an **`all`** AID range for all the ports on a shelf will return the data for the existing OP2G5 circuit packs. RTRV-OC48 with a single AID of a non-OC48 port will fail.

If during the processing of a RTRV command which reports output responses for a number of AIDs an internal error occurs while the system has already started to send response data, the system does not terminate the output response with the ″;″ character.

This behavior will lead to a time-out of the manager and cause him to abort the command.

Related TL1 messages

All other TL1 commands refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for common requirements.

Additional notes

Compliance of output parameter blocks with GR-833

For commands that have output parameter blocks, like the following one. The block may have optional trailing commas on a line.

```
sid date time
```

```
M  ctag COMPLD
   ″aid:<parameter block>″
   ...
;
```

AID grouping of output lines

For commands where the output has the AID prepended to each line, like this:

```
   sid date time
M  ctag COMPLD
   ″aid:block″
   ...
;
```

If the parameters for the same AID value are on separate lines, those lines are grouped together.

spec_blocks continued on another line

For commands that have spec_blocks in the output, like this:

```
   sid date time
M  ctag COMPLD
   ″aid:spec_block″
   ...
;
```

If the output line becomes too long, the keywords may be continued on another line, but the AID is repeated on the continuation line. In addition, the output lines are not limited to 80 characters, and there are no requirements as to how many parameters appear on a line.

Refer to the following example

```
RTRV-EQPT:UNITE-NE:system:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″system:systype=LXC40″
   ″system:istddflt=SDH″
   ″system:trbmddflt=FIXED″
   ″system:synccrtc=SDH″
;
```

References

GR-833-CORE

RTRV-IP-MAP: Retrieve TCP/IP map

The **RTRV-IP-MAP** command retrieves the entries from the IP map.

The *tl1tcpipgw* parameter in the **RTRV-NE** command indicates whether or not the mapping feature is enabled.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-IP-MAP**:tid*::ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.

Output format

If the RTRV-IP-MAP request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   "system::spec_block"
   ...
   "system::spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Output parameters

1. **`system`** – system AID
   This parameter shall denote the system AID of the NE.
   Type: AID
   Range: system (default).
2. **`ip`** – IP Address
   This parameter is used to set or retrieve the IP address(es) of the manager(s) of the NE.
   Type: IPv4 address: four dot separated numeric strings.
   Values: IPv4 address: Each numeric string: 0 .. 255. Number of entries: Mmaximum of 20 entries allowed.
   Default: null.
3. **`acid`** – Application Context ID
   This parameter is used to set or retrieve the application context ID to be assigned to a particular manager IP address.
   Type: Alphanumeric string of length 0 .. 23.
   Range: TL1MAINTENANCE, TL1MEMORYADMINISTRATION, TL1TEST, TL1OTHER1, TL1PEERCOMM.
   Default: TL1PEERCOMM.
   Although acid is a mandatory parameter it will not be evaluated in the NE. The parameter is present for backward compatibility.

Detailed behavior description

The command shall retrieve the spec_block parameters as described in the Output parameter section.

Sorting: If several managers are entered, the system shall respond with one line per *ip* parameter and sort the output according the *ip* parameter.

Example input/output

The following example shows the result of a RTRV-IP-MAP command with one manager entered:

```
RTRV-IP-MAP:LT-UNITE::123456;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
      ″system::ip=198.78.46.8,acid=TL1MAINTENANCE″
      ″system::ip=198.78.46.13,acid=TL1TEST″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`RTRV-NE-IP-MAP`** command.

Related TL1 Messages

- "DLT-IP-MAP: Delete TCP/IP map" (p. 3-198)
- "ENT-IP-MAP: Enter TCP/IP map" (p. 3-1594)

RTRV-L1ADR: Retrieve level 1 Address Data Repository

The command **RTRV-L1ADR** requests information of the level 1 topology data repository, which contains information about available TNAs of a connected multi-domain network. The level 1 topology data repository is only available on ONNS nodes defined as level 1 routing controllers.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: NA

Abortable: No.

Input format

Command syntax: **RTRV-L1ADR**:*srcTid:srcAid:ctag::dataType[,componentID];*

Description

The command **RTRV-L1ADR** retrieves information stored in the ONNS data repository of level 1 routing controllers that is in RAM Memory. The command **RTRV-L1ADR** operates on the current, active-memory-resident, ONNS data. The NVM data from NNDM is used during initialization to recover the ONNS local data repository. Data from other level 1 routing controllers is acquired by the flooding and exchange of LSAs.

Retrieving ALL level 1 address data repository records

When the *dataType* = ALL, Network Navigator shall respond to a request to retrieve its level 1 address data repository by sending a complete report of the contents of the data repository specified. The request shall cause the ONNS node with level 1 routing controller functionality to send all information in the specified data repository.

Retrieving level 1 address data repository records for any routing controller ID

The command **RTRV-L1ADR** with *dataType* = LSAID and *componentID* = L1RCID shall retrieve the L1ADR LSA records of the adjacent routing controllers of neighbor domains related to the specified routing controller with the L1RCID. This L1RCID can be the address of any domain of the connected multi-domain network.

Notification is not generated

The command **RTRV-L1ADR** shall not generate any kind of notification.

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE.
   See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn (default).
3. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ″.″).
   It should be a unique value for each new command.
4. **`dataType`**
   This parameter specifies, which part of the data repository is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range:
   - *ALL*: retrieves the entire (L1TDR) level 1 Topology data repository in RAM
   - *LSAID*: retrieves the LSA in level 1 topology data repository related to the routing controller ID, *componentID* must be a L1RCID
5. **`componentID`**
   componentID depends on the previous *dataType* parameter and is one of:
   - not represented when *datatype* = ALL
   - *L1RCID*: when *datatype* = LSAID
     Description: *routeControlID* defines the identity of the level 1 routing controller
     Type: Four dot separated numeric string

Range: 0 … 255, for each numeric string

Output format

The level 1 Topology data repository (L1ADR) record structure

The complete L1ADR of one routing controller consists of LSARecords for all known routing controllers (including the reporting routing controller itself) in the multi-domain topology. Within a LSARecord all (direct) neighbor domain releationships are kept in TNARecords (one record for each neighbor domain relation).

Level 1 address data repository (L1ADR) without data

An addressed node without data in the L1ADR shall response only the HeaderRecord with *NumberOfLSAs* = 0. The level 1 address data repository record is composed of an *L1ADRHeaderRecord* and multiple *LSARecords* which are described below:

L1ADRRecord ::=

<L1ADRHeaderRecord>

<LSARecord_1>

…

<LSARecord_n>

L1ADRHeaderRecord ::=

″*l1tdrID*,*dataType[=componentID]*,*srcNodeASTNNodeID*, NumberOfLSAs=*LSA_Count*″

LSARecord ::=

<LSAHeaderRecord>

<TNARecord_1>

…

<TNARecord_n>

LSAHeaderRecord ::=

″LSAAdvertisedRouter=*L1RcID*,TNACount=*TNAcount*″

TNARecord ::=

LocAdvNodeID=*locAdvNodeID*,NumberTNAs=*numberTNAs*″

″*TNAtype_1=TNAaddress_1[,TNAtype_2=TNAaddress_2]*″

″*TNAtype_3=TNAaddress_3[,TNAtype_3=TNAaddress_3]*″

…

″*TNAtype_n-1=TNAaddress_n-1[,TNAtype_n=TNAaddress_n]*″

Output format with normal data retrieve completion report for dataType=ALL

If the RTRV-L1ADR request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″L1ADR_HeaderRecord″
   ″LSAHeaderRecord_1″
     ″TNARecord_1_1″
   ...
   ″LSAHeaderRecord_n″
     ″TNARecord_n_1″
     ″TNARecord_n_2″
>
   srcTid date time
M  ctag COMPLD
   ″LSAHeaderRecord_n+1″
     ″TNARecord_n+1_1″
   ...
   ″LSAAddr_HeaderRecord_m″
     ″TNARecord_m_1″
     ″TNARecord_m_2″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the OUTPUT PARAMETERS section of the **RTRV-PATH** command.

L1ADRHeader_Record

1. **l1adrID**
   NN Data Repository dataID to be returned
   Type: Alphanumeric value set.
   Range: L1ADRLevel 1 Address Data Repository in RAM
   Default: L1ADR
2. **dataType**
   This parameter specifies, which part of the data repository is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range: ALL retrieves the entire (L1ADR) level 1 Topology data repository in RAM. LSAID retrieves the LSA in level 1 topology data repository related to the routing controller ID, *componentID* must be a L1RCID.
3. **componentID**
   componentID depends on the previous *dataType* parameter and is one of:
   - not represented when *datatype* = ALL
   - *L1RCID*: when *datatype* = LSAID

Description: *routeControlID* defines the identity of the level 1 routing controller
Type: Four dot separated numeric string
Range: 0 … 255, for each numeric string
Note: If *dataType* = ALL, *componentID* and the "=" sign are not reported.

4. **`srcNodeASTNNodeID`** – Reporting Node ID
   ASTN node ID of the ONNS node replying to this command.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0 … 255

LSARecord

For each reported primary routing controller of the level 1 topology one LSARecord is reported.

1. **`LSAHeaderRecord`**
   All parameters in the LSAHeaderRecord describe properties of the *LSAAdvertisedRouter*.
   L1RcID – ID of Advertising Routing Controller
   Description: *routeControlID* defines the identity of the level 1 routing controller
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
2. **`TNARecord`**
   localAdvNodeID – Local Advertised Node ID
   The *localAdvNodeID* of the advertiser of the LSA with the address data following in this TNARecord.
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
   numberTNAs
   Number of reported TNAs known in the advertising level 1 node.
   Type: Integer
   Range: ≥0
   TNAType
   TNAType specifies the address type of the TNA address.
   Type: Alphanumeric value set.
   Range:
   - IPV4TNAADDR (default): TNA address is a IPv4 address
   - IPV6TNAADDR: TNA address is a IPv6 address
   - NSAPTNAADDR: TNA address is a NSAP address

TNAAddress depends on the previous *TNAtype* parameter and is one of:

- IPv4TNAAddress (when TNAtype=IPV4TNAADDR)
- IPv6TNAAddress (when TNAtype=IPV6TNAADDR)
- NSAPTNAAddress (when TNAtype=NSAPTNAADDR)

IPv4TNAAddress
Identifier of a TNA address in IPv4 format.
Type: four dot separated numeric string
Range: 0 … 255, for each numeric string

IPv6TNAAddress
Identifier of a TNA address in IPv6 format.
Type: Eight hyphen separated hexadecimal numbers
Range: for each hexadecimal number: 0..FFFF

NSAPTNAAddress
Identifier of a TNA address in NSAP format.
Type: Alphanumeric value set (20 byte)
Range: up to 20 alphanumeric strings with hexadecimal values 00..FF

Example input/output

The **RTRV-L1ADR** command with *dataType* = ALL produces this example output result:

```
RTRV-L1ADR:LT-ONNS-61::123456::ALL;

LT-ONNS-61 2006-10-04 09:45:35
M  123456 RTRV
"L1ADR,ALL,135.13.44.61,"
"LSAAdvertisedRouter=218.3.45.56,"
"LocAdvNodeID=128.32.14.32,"
"IPV4TNAADDR=135.13.3.119,IPV6TNAADDR=
3ED-2FC2-3-F365-F290-119-45-A360"
    "LSAAdvertisedRouter=139.38.28.84,"
"LocAdvNodeID=148.43.56.95,"
"NSAPTNAADDR=13F323EFCA45DABC34698723FEAC45CF289364ED"
>
LT-ONNS-61 2006-10-04 09:45:35

M  123456 COMPLD
    "LSAAdvertisedRouter=129.34.93.2,"
"LocAdvNodeNodeID=118.23.59.15,"
"IPV4TNAADDR=176.113.5.119,IPV4TNAADDR=135.16.3.120,IPV6TNAADDR=
1ED-2FC2-3-F365-E290-119-45-A321"
;
```

Error responses

No Routing Controller available

The command is rejected when no routing controller is available on the addressed NE.

Error code: `IDNV – Input, Data Not Valid, RC node not created`

Invalid *componentID* parameter

The command is rejected when the received *componentID* is invalid.

Error code: `IRCI – Input, Invalid Routing Controller ID`

Related TL1 messages

- “ACT-L1ROUTING” (p. 3-15)
- “CANC-L1ROUTING” (p. 3-54)
- “ENT-CONN: Establish network connections with TNA address parameters” (p. 3-1455)
- “ENT-CONN: Establish network connections with TNA address parameters” (p. 3-1455)
- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “REPT CONNCHG notification” (p. 3-2104)
- “RTRV-NADR: Retrieve ONNS network (domain) address data repository” (p. 3-2539)

RTRV-L1TDR: Retrieve level 1 topology data repository

The command **RTRV-L1TDR** requests information of the level 1 topology data repository, which contains information about abstract nodes, abstract links of a connected multi-domain network. The level 1 topology data repository is only available on ONNS nodes defined as level 1 routing controllers.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: NA

Abortable: No.

Input format

Command syntax: **RTRV-L1TDR**:*srcTid:srcAid:ctag::dataType[,componentID];*

Description

The command **RTRV-L1TDR** retrieves information stored in the ONNS data repository of level 1 routing controllers that is in RAM Memory. The command **RTRV-L1TDR** operates on the current, active-memory-resident, ONNS data. The NVM data from NNDM is used during initialization to recover the ONNS local data repository. Data from other level 1 routing controllers is acquired by the flooding and exchange of LSAs.

Retrieving ALL level 1 topology data repository records

When the *dataType* = ALL, Network Navigator shall respond to a request to retrieve its level 1 topology data repository by sending a complete report of the contents of the data repository specified. The request shall cause the ONNS node with level 1 routing controller functionality to send all information in the specified data repository.

Retrieving level 1 topology data repository records for any routing controller ID

The command **RTRV-L1TDR** with *dataType* = LSAID and *componentID* = L1RCID shall retrieve the L1TDR LSA records that relate to all of the links from the specified routing controller with the L1RCID to adjacent routing controllers of neighbor domains. This RCID can be the address of any domain of the connected multi-domain network.

Notification is not generated

The command **RTRV-L1TDR** shall not generate any kind of notification.

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE.
   See also the definition of *tid* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Type: AID
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn (default).

3. **`ctag`**
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   See also the definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).
   Type: Alphanumeric string
   Range: The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing "*.*").
   It should be a unique value for each new command.

4. **`dataType`**
   This parameter specifies, which part of the data repository is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range:
   - *ALL*: retrieves the entire (L1TDR) level 1 Topology data repository in RAM
   - *LSAID*: retrieves the LSA in level 1 topology data repository related to the routing controller ID, *componentID* must be a L1RCID

5. **`componentID`**
   componentID depends on the previous *dataType* parameter and is one of:
   - not represented when *datatype* = ALL
   - *L1RCID*: when *datatype* = LSAID
     Description: *routeControlID* defines the identity of the level 1 routing controller
     Type: Four dot separated numeric string

Range: 0 … 255, for each numeric string

Output format

The level 1 Topology data repository (L1TDR) record structure

The complete L1TDR of one routing controller consists of LSARecords for all known routing controllers (including the reporting routing controller itself) in the multi-domain topology. Within a LSARecord all (direct) neighbor domain releationships are kept in LinkRecords (one record for each neighbor domain relation).

Level 1 Topology data repository (L1TDR) without data

An addressed node without data in the L1TDR shall response only the HeaderRecord with *NumberOfLSAs* = 0. The level 1 topology data repository record is composed of an *L1TDRHeaderRecord* and multiple *LSARecords* which are described below:

L1TDRRecord ::=

<L1TDRHeaderRecord>

<LSARecord_1>

…

<LSARecord_n>

L1TDRHeaderRecord ::=

″*l1tdrID,dataType[=componentID],srcNodeASTNNodeID*, NumberOfLSAs=*LSA_Count*″

LSARecord ::=

<LSAHeaderRecord>

<LSARecord_1>

…

<LSARecord_n>

LSAHeaderRecord ::=

″LSAAdvertisedRouter=*L1RcID*,LinkCount=*linkCount*″

LinkRecord ::=

″LocNodeID=*locNodeID*,RemoteNodeID=*remoteNodeID*,

LocLinkID=*locLinkID*,RemoteLinkId=*remoteLinkID*,

LinkType=*linkType*,LinkID=*linkID*,LinkMetric=*adminCost*,

SRLG=*srlgList*,LinkResourceClass=*linkReCls*,

LinkProtType=*linkProtType*,SwitchCapDesc=*bandwidthList*″

Output format with normal data retrieve completion report for dataType=ALL

If the RTRV-L1TDR request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″L1TDR_HeaderRecord″
   ″LSAHeaderRecord_1″
     ″LinkRecord_1_1″
     ″LinkRecord_1_2″
   ...
   ″LSAHeaderRecord_n″
     ″LinkRecord_n_1″
     ″LinkRecord_n_2″
     ″LinkRecord_n_3″
>
   srcTid date time
M  ctag COMPLD
   ″LSAHeaderRecord_n+1″
     ″LinkRecord_n+1_1″
     ″LinkRecord_n+1_2″
     ″LinkRecord_n+1_3″
     ″LinkRecord_n+1_4″
   ...
   ″LSAAddr_HeaderRecord_m″
     ″LinkRecord_m_1″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the OUTPUT PARAMETERS section of the **RTRV-PATH** command.

L1TDRHeader_Record

1. **l1tdrID**
   NN Data Repository dataID to be returned
   Type: Alphanumeric value set.
   Range: L1TDRLevel 1 Topology data repository in RAM
   Default: L1TDR
2. **dataType**
   This parameter specifies, which part of the data repository is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range: ALL retrieves the entire (L1TDR) level 1 Topology data repository in RAM. LSAID retrieves the LSA in level 1 topology data repository related to the routing controller ID, *componentID* must be a L1RCID.
3. **componentID**

componentID depends on the previous *dataType* parameter and is one of:

- not represented when *datatype* = ALL
- *L1RCID*: when *datatype* = LSAID
  Description: *routeControlID* defines the identity of the level 1 routing controller
  Type: Four dot separated numeric string
  Range: 0 … 255, for each numeric string
  Note: If *dataType* = ALL, *componentID* and the "=" sign are not reported.

4. **`srcNodeASTNNodeID`** – Reporting Node ID
   ASTN node ID of the ONNS node replying to this command.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0 … 255

LSARecord

For each reported primary routing controller of the level 1 topology one LSARecord is reported.

1. **`LSAHeaderRecord`**
   All parameters in the LSAHeaderRecord describe properties of the *LSAAdvertisedRouter*.
   L1RcID – ID of Advertising Routing Controller
   routeControlID defines the identity of the level 1 routing controller
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string

2. **`LinkRecord`**
   locNodeID
   The *locNodeID* is the representation of a physical node of the local domain for the level 1 routing.
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
   remoteNodeID
   The *remoteNodeID* is the representation of a physical node of a neighbor domain for the level 1 routing.
   Type: four dot separated numeric string
   Range: 0 … 255, for each numeric string
   locLinkID
   The *locLinkID* specifies from the local domain's point-of-view the identification for the link the corresponding port is assigned to.
   Type: Numeric string.
   Range: 0..42946967295
   remoteLinkID
   The *remoteLinkID* specifies from the neighbor domain's point-of-view the identification for the link the corresponding port is assigned to.
   Type: Numeric string.
   Range: 0..42946967295
   linkType

The *linkType* defines the type of the link.
Type: Alphanumeric value set.
Range: PtP (point to point)

linkID

The *linkID* identifies the level 1 Routing Controller at the other end of the link.
Type: four dot separated numeric string
Range: 0 … 255, for each numeric string

adminCost – Administrative Cost

Provisioned administrative cost of the port connection when used in a connection path. See also parameter *adminCost* in RTRV-rr.
Type: Integer
Range: 1 (default) … 65535
Administrative cost parameter is not applicable for EDGE and UNI ports and is set to the default value.

srlgList

Shared Risk Link Group Identifier list.
See also parameter *srlgList* in RTRV-rr.
The max. number of identities that can be provisioned per port may be release dependent.
Type: Up to ten &-separated Integers
Examples:
198&233&0&0&0&0&0&0&0&0
10&20&30&40&0&0&0&0&=&0&0
89&1000&4&10&20&31&111&231&300&777
Range: 0 … 65535
Default. 0 (means no SRLG provisioned)
SRLG list is not applicable for EDGE and UNI ports and is set to the default value.

linkResCls

The *linkResCls* globally classifies routing purposes the link of the corresponding port is assigned to.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF

linkProtType

The *linkProtType* globally classifies routing purposes the link of the corresponding port is assigned to.
Type: Set of alphanumeric values, separated by “&”.
Range for each alphanumeric value:

- NONE: value specifying that no link protection type is set
- ET: value specifying that link is for extra traffic
- 0x1: value specifying that “unprotected” link protection type is set
- 1:1-W: value specifying that “1:1” link protection type is set
- 1+1-W: value specifying that “1+1” link protection type is set
- ENH: value specifying that an enhanced protection type is set

- RESV1: reserved value for further use)
- RESV2: reserved value for further use

Examples:

- “0x1&1+1-W” will be reported if both “unprotected” and “1+1” are set.
- “NONE” will be reported only without additional values.

Note:
Multiple values or values different from “0x1” and “1+1-W” will only occur for links between domains from “foreign” vendors, as for “own” links either “0x1” or “1+1-W” will only be specified.

bandwidthList

The *bandwidthList* indicates the available bandwidth per signal rate.
Type: Character string.
Range:

Bandwidth List: five dash separated numeric strings representing STS1 #, STS3c #, STS12c #, STS48c #, STS192c # with:

- STS1 #0..4097
- STS3c #0..1023
- STS12c #0..255
- STS48c #0..63
- STS192c #0..15

Example for an available bandwidth of 2 STS48c and a defined minimal XC STS3c: 0-32-8-2-0.

Example input/output

The **RTRV-L1TDR** command with *dataType* = ALL produces this example output result:

```
RTRV-L1TDR:LT-ONNS-61::123456::ALL;

LT-ONNS-61 2006-10-04 09:45:35
M  123456 RTRV
″L1TDR,ALL,135.13.44.61,″
″LSAAdvertisedRouter=218.3.45.56,″
    ″LocNodeID=128.32.14.32,RemoteNodeID=148.43.56.95,
LocLinkID=118.23.95.83,RemoteLinkId=184.71.75.23,LinkType=PtP,LinkID=
139.38.28.84,LinkMetric=24,SRLG=0&0&0&0&0&0&0&0&0&0,LinkResourceClass=
1FC123A2,LinkProtType=0x1,SwitchCapDesc=0-16-4-1-0″
″LSAAdvertisedRouter=139.38.28.84,″
      ″LocNodeID=148.43.56.95,RemoteNodeID=128.32.14.32,LocLinkID=
184.71.75.23,RemoteLinkId=118.23.95.83,LinkType=PtP,LinkID=218.3.45.56,
LinkMetric=24,SRLG=0&0&0&0&0&0&0&0&0&0,LinkResourceClass=1FC423B2,
LinkProtType=0x1,SwitchCapDesc=0-16-4-1-0″
″LocNodeID=148.43.56.95,RemoteNodeID=118.23.59.15,
LocLinkID=23.95.19.45,RemoteLinkId=123.95.82.4,LinkType=PtP,LinkID=
129.34.93.2,LinkMetric=260,SRLG=1&34&0&0&0&0&0&0&0&0,LinkResourceClass=
2FD125C2,LinkProtType=1+1,SwitchCapDesc=0-4-1-0-0″
```

```
>
LT-ONNS-61 2006-10-04 09:45:35

M  123456 COMPLD
"LSAAdvertisedRouter=129.34.93.2,"
      "LocNodeID=118.23.59.15,RemoteNodeID=148.43.56.95,
LocLinkID=123.95.82.4,RemoteLinkId=23.95.19.45,LinkType=PtP,LinkID=
139.38.28.84,LinkMetric=260,SRLG=1&34&0&0&0&0&0&0&0&0,LinkResourceClass=
00000000,LinkProtType=1+1,SwitchCapDesc=0-4-1-0-0"
;
```

Error responses

IDNV - Input, Data Not Valid, RC node not created

The command is rejected when no routing controller is available on the addressed NE.

Error code: `INRC – Input, No Routing Controller`

Invalid *componentID* parameter

The command is rejected when the received *componentID* is invalid.

Error code: `IRCI – Input, Invalid Routing Controller ID`

Related TL1 messages

- "ACT-L1ROUTING" (p. 3-15)
- "CANC-L1ROUTING" (p. 3-54)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
- "DLT-CONN: Delete network connection" (p. 3-139)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT CONNCHG notification" (p. 3-2104)
- "RTRV-ELDR: Retrieve ONNS external link data repository" (p. 3-2339)

RTRV-LOG-ALM: Retrieve log alarm

The **RTRV-LOG-ALM** command retrieves the alarm log.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-LOG-ALM**:*tid*::*ctag*::[*startdate*],[*starttime*],[*replytype*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **startdate** – Start Date
   The date for the starting point of the report. If both startdate and starttime are omitted, the entire log is retrieved. Both startdate and starttime must be specified, or both must be omitted.
   Type: value set of the format YY-MM-DD.
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
4. **starttime** – Start Time
   The time for the starting point of the report. If both startdate and starttime are omitted, the entire log is retrieved. Both startdate and starttime must be specified, or both must be omitted.
   Type: value set of the format HH-MM-SS.
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents the hours, MM the minutes, SS the seconds.
5. **replytype** – Type of Reply
   This parameter defines the kind of log retrieving.
   Type: Alphanumeric value set.

   Range:

   - BROWSE (User Browsing) – If the requested *startdate* and *starttime* is in the log, those entries and any subsequent log entries are returned. If the requested *startdate* and *starttime* is NOT in the log, the first log entry in the log with a timestamp later than the *startdate* and *starttime* plus subsequent log entries will be returned.

Important: If there are no alarm entries in the specified interval the system returns with the error code
`SNVS /* Status, Not in Valid State, no matching log entry */.`
Instead of this an empty log with a COMPLD message should be returned.

- DBSYNC – This value provides the same function as BROWSE. The value DBSYNC is allowed for compatibility reasons.

Default: BROWSE, if the parameter is omitted.

Output format

Normal completion response of successful session termination

The following example shows the successful completion of the **`RTRV-LOG-ALM`** command by the NE:

```
   sid date time
M  ctag COMPLD

      "aid,aidtype:[ntfcncde],condtype,[srveff],ocrdat,ocrtm,[condeff][:conddescr]"

        ...
   "aid,aidtype:[ntfcncde],condtype,[srveff],ocrdat,ocrtm,[condeff][:conddescr]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This is the address of the entity for which an alarm condition is being reported.
   Type: AID.
   Range: All defined AIDs are allowed. It should be a valid AID for the specified *modifier*. Refer to the definition of AIDs.
2. **`aidtype`** – Access Identifier Type
   This parameter defines the AID type to which the alarm belongs.
   Type: Alphanumeric value set.
   Range:
   - COM: Common. An alarm/event that applies to the whole NE system; for example, a processing error.
   - EQPT: An equipment-related alarm/event.
   - SYST: An alarm/event related to system timing.
   - OUTT: An alarm/event related to output timing.

- STM1E: A facility-related alarm/event for synchronous transport mode 1 (electrical).
- STM1: A facility-related alarm/event for synchronous transport mode 1.
- STM4: A facility-related alarm/event for synchronous transport mode 4.
- STM16: A facility-related alarm/event for synchronous transport mode 16.
- STM16T: A facility-related alarm/event for fully transparent synchronous transport mode 16.
- STM64: A facility-related alarm/event for synchronous transport mode 64.
- STM256: A facility-related alarm/event for synchronous transport mode 256.
- ODU1: A facility-related alarm/event for optical data units of level 1
- OC3: A facility-related alarm/event at the optical carrier signal level 3.
- OC12: A facility-related alarm/event at the optical carrier signal level 12.
- OC48: A facility-related alarm/event at the optical carrier signal level 48.
- OC48T: A facility-related alarm/event at the fully transparent optical carrier signal level 48.
- OC192: A facility-related alarm/event at the optical carrier signal level 192
- OC768: A facility-related alarm/event at the optical carrier signal level 768
- T3: A facility-related alarm/event at the DS3 level
- TMAT3: A facility-related alarm/event at the DS3 level related to TransMUX function
- EC1: A facility-related alarm/event at the EC1 level
- 1GE: A facility-related alarm/event for 1 Gigabit Ethernet.
- 1GEPL: A facility-related alarm/event for 1 Gigabit Ethernet Private Line.
- 10GEPL: A facility-related alarm/event for 10 Gigabit Ethernet Private Line.
- VC3: A facility-related alarm/event for virtual container 3.
- VCG: A facility-related alarm/event for virtual concatenation group.
- VC4: A facility-related alarm/event for virtual container 4
- VC44C: A facility-related alarm/event for a signal concatenated at 4 times the VC4 rate.
- VC416C: A facility-related alarm/event for a signal concatenated at 16 times the VC4 rate.
- VC464C: A facility-related alarm/event for a signal concatenated at 64 times the VC4 rate.
- STS1: A facility-related alarm/event at synchronous transport signal level 1.
- TMASTS1: A facility-related alarm/event at synchronous transport signal level 1 related to TransMUX function.
- STS3C: A facility-related alarm/event at synchronous transport signal level 3.

- STS12C: A facility-related alarm/event at synchronous transport signal level 12.
- STS48C: A facility-related alarm/event at synchronous transport signal level 48.
- STS192C: A facility-related alarm/event at synchronous transport signal level 192.
- HOVC4 : A facility-related alarm/event for virtual container 4 higher-order path termination.
- HOSTS1: A facility-related alarm/event at synchronous transport signal level 1 higher-order path termination.
- LOVC3: A facility-related alarm/event for virtual container 3 (lower-order path).
- LOVC12: A facility-related alarm/event for virtual container 12
- LOVT1: A facility-related alarm/event at transport level VT1.5.
- TMSVT: A facility-related alarm/event at transport level VT1.5 related to TransMUX function.
- DS1: A facility-related alarm/event at transport level DS1.
- LINK: A signaling communication link related alarm/event for ONNS

Refer to “Alarms, events, ASAPs” (p. 6-4) for the relation of alarm / entity / *aidtype*.

3. **`ntfcncde`** – Notification Code
   This is the alarm severity (alarm level).
   Type: Alphanumeric value set.
   Range:
   - CR – Critical (ITU / Telcordia)
   - MJ – Major (ITU / Telcordia)
   - MN – Minor (ITU / Telcordia)
   - SC – Standing condition – Warning (ITU) / not alarmed (Telcordia)
   - CL – Cleared

   The abnormal condition is always a standing condition, this means it has a beginning and an end (SC and CL). The severity of a abnormal condition is not provisionable. It is always NA – warning (ITU) / not alarmed (Telcordia) which implies ABN messages are always reported via REPT EVT.
4. **`condtype`** – Condition Type
   This parameter identifies the probable cause (type of alarm indication) being reported.
   Type: Alphanumeric value set.
   Range: All defined probable causes.
   For ABN conditions see definition of *condtype* in the output parameters section of “REPT EVT: Report event” (p. 3-2121).

5. **`srveff`** – Service Effect
   This parameter identifies the service affect of alarms being reported. It presents the "best guess" from a local system point of view.
   Type: Alphanumeric value set.
   Range:
   - SA: Service-Affecting alarm condition
   - NSA: Non-Service-Affecting alarm condition.

   Abnormal condition event messages are always handled as NSA.
6. **`ocrdat`** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format YY-MM-DD.
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
7. **`ocrtm`** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
8. **`condeff`** – Condition Effect
   This indicates the effect of the reported event on the network element. It shall be used to distinguish between persistent and transient alarms/events.
   Two state transitions can be distinguished for a persistent alarm/event:
   - raise transition: state change from not present to present
   - clear transition: state change from present to not present

   A transient alarm/event has no "presence" state; that means, just the occurrence of that event/alarm is reported.
   ABN conditions are always persistent and not transient.
   Type: Alphanumeric value set.
   Range:
   - CL: Standing condition cleared for a persistent alarm/event
   - SC: Standing condition raised for a persistent alarm/event
   - TC: Transient condition occurred for a transient alarms/events

   Default: None.
9. **`conddescr`** – Description of the Condition
   This is a textual description of the condition.
   Type: inner string of the format: "<category>, <entity>, <description>, <additional text>" or "<category>, <entity>, <description>, <additional text>-cleared" (this form is used in REPT ALM/EVT for cleared alarms).

Range: :

- category
- entity: the entity for which the alarm is reported
- description: the alarm description, the string is up to 32 characters
- additional text. If there is no additional text, then the trailing comma may be omitted.

Default: None.
Range:
Description for a protection group AID:

- Abnormal condition, forced switch active
- Abnormal condition, lockout active

Description for a timing reference AID:

- Abnormal condition, operate in hold-over
- Abnormal condition, operate in free run mode

Description for a port AID:

- Abnormal condition, facility loopback active

Description for a logical tributary AID:

- Abnormal condition, cross-connect loopback active
- Abnormal condition, test access session active

Description for a system AID:

- Abnormal condition, System operates in maintenance mode

Detailed behavior description

This log gives the user a history and time sequence of all of the most recent alarm detected by the system regarding equipment or facilities attached to the system.

TCAs are not stored within the alarm log.

Storage Order Dependency: If the storage order of entries in the alarm log is not in line with the occurrence date and time, then the command may retrieve entries which are older than the specified date and time. A storage order, which is not in line with the date and time of the entries can occur after setting the system date and time.

The **RTRV-LOG-ALM** command does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful completion of the **RTRV-LOG-ALM** command by the NE:

```
RTRV-LOG-ALM:LT-UNITE-123::123456::;

   LT-UNITE-123 00-01-01 08:00:00
```

```
M  123456 COMPLD
   ″1-1-#-#-3-1,OC48:CR,LOS,SA,01-01,07-29-13:\″Communication-Transport, OC48 port,
      LOS of Signal\″″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-LOG-ALM** command.

Related TL1 messages

- “RTRV-ALM: Retrieve alarm” (p. 3-2231)
- “RTRV-LOG-NTFCN: Retrieve notification log” (p. 3-2464)

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: NE and Transport Surveillance Messages.

RTRV-LOG-NNEVT: Retrieve NN event log

The **RTRV-LOG-NNEVT** command retrieves the NN event log which is used to store the most recent NN events non volatile.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-LOG-NNEVT**:*tid::ctag::[startdate],[starttime];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
3. **startdate** – Date for the Starting point
   The format is YY-MM-DD. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set.

   Range:

   - for YY: 00 to 99
   - for MM: 01 to 12
   - for DD: 01 to 31.
4. **starttime** – Time for the Starting point
   The format is HH-MM-SS. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set.

Range:

- for HH: 00 to 23
- for MM: 00 to 59
- for SS: 00 to 59.

Output format

If the RTRV-LOG-NNEVT request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″Time=time,Date=date:nnevt″
    ...
   ″Time=time,Date=date:nnevt″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

1. **`date`** – Date
   This parameter shows the occurrence date of the NN event. .
   Type: Value set.
   Range: YY-MM-DD
2. **`time`** – Time
   This parameter shows the occurrence time of the NN event. .
   Type: value set.
   Range: HH-MM-SS.
3. **`nnevt`** – NN Event
   This parameter shall provide the details of the ONNS event. It consists of the text block without the date/time information of the ONNS event enclosed in double quotes (″).
   Note: The following ONNS event types are stored in this log:
   REPT PTHCHG
   REPT PTHEVT
   REPT NNCHG
   REPT NNEVT
   REPT CONNCHG
   The detailed TL1 specification can be found in SRD_TL1_ONNS.
   Type: inner string.
   Range: a string of arbitrary length, any character except ″ and \, upper and lower cases are considered.

Detailed behavior description

If the error log of the system does not contain any records at all or no records exist matching the specified start date and start time, an empty reply is returned. Note: The log contains the last 5000 ONNS events.

Example input/output

The following is an example of the **RTRV-LOG-NNEVT** command:

```
RTRV-LOG-NNEVT:LT-UNITE::123456::;

   LT-UNITE 02-02-05 08:00:00
M  123456 COMPLD
    ″TIME=07-59-59,DATE=02-07-01:CCB=PATH:125.114.48.12:SNN:C015:\″ Connection
      restored and Saved in NVM \″:″
   ″ConnID=125.114.48.12&1-1-3-4-1&1,ConnStatus=CON&OK,2WAY,AUTONR,STS192,
   connAlias=AEXP-NY-BOS-ckt,
   MaxDelay=199,,,SrlgExclude=630&820&720,IncludeID=0,RstrPriority=3,,
   SrcNID=125.114.48.12,1-1-3-4-1,DestNID=125.6.19.43,1-1-14-2-1:
   PathID=125.114.48.12-12123,State=CON &OK,
   NodeList=125.114.48.12&nodeIP1&nodeIP2&...&125.6.19.43:
   PathID=125.114.48.12-11013,State=CON &TRDOWN,
   NodeList=125.114.48.12&nodeIP4&nodeIP5&...&125.6.19.43″
   . . .
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-LOG-NNEVT** command.

RTRV-LOG-NTFCN: Retrieve notification log

The **`RTRV-LOG-NTFCN`** command retrieves the notification log, which is used to store the most recent notification reported via **`REPT DBCHG`**. This log allows the user to see the most recent activity that has caused updates to the system database. This is useful in solving database audit problems. This information can also be used to re-synchronize a management system with the NE after a communications outage between the two.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **`RTRV-LOG-NTFCN`**:*tid*::*ctag*::*[startdate]*,*[starttime]*,*[replytype]*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **`startdate`** – Date for the Starting point
   The format is YY-MM-DD. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set.
   Range:
   - for YY: 00 to 99
   - for MM: 01 to 12
   - for DD: 01 to 31.
4. **`starttime`** – Time for the Starting point
   The format is HH-MM-SS. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set.

Range:

- for HH: 00 to 23
- for MM: 00 to 59
- for SS: 00 to 59.

5. **`replytype`** – Type of Reply
BROWSE is used if user browsing is needed. If the requested *startdate* and *starttime* is in the log, those entries and any subsequent log entries are returned. If the requested *startdate* and *starttime* is NOT in the log, the first log entry in the log with a timestamp later than the *startdate* and *starttime* plus subsequent log entries will be returned. If the requested *startdate* and *starttime* is NOT in the log only a completion message is returned without further error messages.
Important: If there are no alarm entries in the specified interval the system returns with the error code
`SNVS /* Status, Not in Valid State, no matching log entry */`. Instead of this an empty log with a COMPLD message should be returned.
DBSYNC is used if the management system requires the log for database synchronization. If the requested *startdate* and *starttime* is in the log, those entries and any subsequent log entries are returned. If the requested *startdate* and *starttime* is NOT in the log, an SNVS error response is returned.
Type: Alphanumeric value set.
Range:
   - BROWSE
   - DBSYNC.

   Default: BROWSE.

Output format

If the RTRV-LOG-NTFCN request completes successfully, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″Time=time,Date=date:[ccb]:[aid]:[com_block]:[spec_block]″
   [″aux_block″]
   [″aux_block″]
        ...
   ″Time=time,Date=date:[ccb]:[aid]:[com_block]:[spec_block]″
   [″aux_block″]
   [″aux_block″]
        ...
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

1. **`date`** – Date
   Type: Value set.
   Range: YY-MM-DD
2. **`time`** – Time
   Type: value set.
   Range: HH-MM-SS.
3. **`ccb`** – Command Code Block
   This field will be set to the TL1 command that causes that database change. The *com_block* and *spec_block* fields will be set according to the parameters that changed.
   Type: Alphanumeric value set verb-modifier[-modifier].
4. **`aid`** – Access Identifier
   AID of the entity for which the change is reported
   Type: AID.
   Range: As defined by the *aid* input parameter of the corresponding command, defined by *ccb*.
5. **`com_block`** – Common Block
   This is a position-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter values are included in this block in the same order as the corresponding TL1 command.
   As exception to this rule are some commands in the protection fragment, where the aliases *rid* and *ppgname* are not reported in the **`REPT DBCHG`** notification.
   If the database change is the result of a TL1 command, then the parameter *ctag* from the TL1 command is not included in the *com_block*.
   Type: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Range: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
   Default: As defined in the input *com_block* of the corresponding command, defined by *ccb*.
6. **`spec_block`** – Specific Block
   This is a name-defined field. If the database change is the result of a TL1 command which contains parameters in this block, the parameter names and values are included in this block. parameters within the specific block are positionally independent and are specified using a name-defined construct of *parameter* = value in a comma-separated list. The parameter names are the same as those for the corresponding TL1 command.
   Type: As defined in the input *spec_block* of the corresponding command, defined

by *ccb*.
Range: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.
Default: As defined in the input *spec_block* of the corresponding command, defined by *ccb*.

7. **aux_block** – Auxiliary Block
The auxiliary block is optional. The format of the auxiliary block depends on the notification which caused the log entry.
Auxiliary block for path protection group related entries
In the case of a path protection group related notification, the auxiliary block shall have the format of the CML Block. The CML block format is defined in the output format definition of the ENT-PROTN-GRP notification and the ED-PROTN-GRP notification.

Example input/output

The following is an example of the **RTRV-LOG-NTFCN** command:

```
RTRV-LOG-NTFCN:LT-UNITE::123456::;

   LT-UNITE 00-11-15 08:00:00
M  123456 COMPLD
   "TIME=07-59-59,DATE=00-11-15:ENT-EQPT:1-1::shlftp=DUR"
   ...
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-LOG-NTFCN** command.

Validation rule: matching log entry

There is at least one matching log entry for the specified *startdate* and *starttime* if reply type equals DBSYNC.

Error code: `SNVS – Status, Not in Valid State, no matching log entry`

Related TL1 messages

“REPT DBCHG: Report data base change” (p. 3-2118)

References

GR-833

RTRV-LOG-PROTNSW: Retrieve log protection switch

The **RTRV-LOG-PROTNSW** command retrieves the protection switch log. This log gives a complete time-ordered list of switching activity which has occurred at the NE covering all different protection groups, system timing reference, station clock output timing reference and system clock.

The protection switch log is a circular buffer that should hold at least 500 entries.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-LOG-PROTNSW**:*tid*::*ctag*::[*startdate*],[*starttime*],[*replytype*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **startdate** – Start Date
   This parameter specifies a date used as filter criteria in order to determine the set of switch events being returned.
   See description of parameter *replytype* for more precise semantics definition.
   Type: value set of format YY-MM-DD
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the year, MM represents month, DD the day.

4. **`starttime`** – Start Time
   This parameter specifies a time used as filter criteria in order to determine the set of switch events being returned.
   See description of parameter *replytype* for more precise semantics definition.
   Type: value set of format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.

5. **`replytype`** – Type of Reply
   This parameter specifies, how the entries in the reply are determined. The value BROWSE is used in case of user browsing. The value DBSYNC is used in case of data synchronization with management system.
   BROWSE semantics: If the requested *startdate* and *starttime* is in the log, those entries with matching timestamp and any subsequent log entries are returned. If the requested *startdate* and *starttime* is NOT in the log, the first log entry in the log with a timestamp later than the *startdate* and *starttime* plus subsequent log entries are returned.
   Important: If there are no alarm entries in the specified interval the system returns with the error code
   `SNVS /* Status, Not in Valid State, no matching log entry */` . Instead of this an empty log with a COMPLD message should be returned.
   DBSYNC semantics: If the requested *startdate* and *starttime* is in the log (i.e. a matching timestamp was found), those entries and any subsequent log entries are returned. If the requested *startdate* and *starttime* is NOT in the log (i.e. a no matching timestamp was found), an SNVS error response is returned.
   Type: Alphanumeric value set.
   Range: BROWSE, DBSYNC.
   Default: BROWSE.

Output format

The returned log entries are ordered ascending according to their timestamp. After receiving the **`RTRV-LOG-PROTNSW`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″aid,aidtype:EVT,condtype,condeff,ocrdat,ocrtm[:\″conddescr\″]″
        <0 or more of the line above>
   ″aid:SW,,,ocrdat,ocrtm″
        <0 or more of the line above>
        <0 or more repetitions of this entire block>
;
```

The output parameters of the log entry are derived/correspond from/to the parameters in the protection switch notification. The parameter definition will reference to the corresponding parameter in the protection switch notification section. Log entry parameters are not splitted within this section w.r.t. protection type. Therefore the protection type has to be considered in order to look-up the proper definition.

Output parameters

Command parameters

1. **`aid`** – Access Identifier
2. **`aidtype`** – Access Identifier Type
3. **`condtype`** – Condition Type
4. **`condeff`** – Condition Effect
   This parameter indicates the effect of the reported event on the network element. The value TC (Transient Condition) is used for all protection switch events. Type alphanumeric value set Range TC
   Type: Alphanumeric value set.
   Range: TC
   a. ocrdat – Occurrence Date
      The parameter shall indicate the date of the switch event being reported. Type: value set of the format YY-MM-DD. Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
   b. ocrtm – Occurrence Time
      This parameter indicates the time of the switch event being reported. Type: value set of format HH-MM-SS. Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
5. **`conddescr`** – Condition Description

Detailed behavior description

If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted. If *replytype* is specified, then *startdate* and *starttime* are present.

The log shows the same information as it could be derived from looking on the protection switches reported by means of REPT EVT and REPT SW.

Note: In case of a 1+1 equipment protection switching causing one REPT SW notification and one REPT EVT notification, then two entries are reported in the retrieve log reply.

Example input/output

1+1 Equipment protection group

The following example retrieves a manual switch to protection circuit pack for 1+1 equipment protection group:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-xcp-cp:SW,,,01-10,08-12-00″
   ″1-1-estgrp,COM:EVT,GP,TC,01-10,08-12-00:\″Protection Switch, EQPT, Manual
      switch\″″
;
```

Multiplex section/line protection group (1+1 and 1:1)

The following example retrieves a lockout switch for a 1+1 multiplex section/line protection group:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-2,OC3:EVT,WKSWBK,TC,01-10,08-12-00:\″Protection Switch, OC3 port,
      Lockout switch,, 1+1/1x1\″″
;
```

2 fibre MS-SPRing/BLSR protection group

The following example retrieves a manual ring switch of the east side for a 2F MS-SPRing/BLSR protection group:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-2,OC48:EVT,WKSWPR-2,TC,01-10,08-12-00:\″Protection Switch, OC48 port,
       Manual switch, MSR, 2F\″″
;
```

4 fibre MS-SPRing/BLSR protection group

The following example retrieves a manual ring switch of the east side for a 4F MS-SPRing/BLSR protection group:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-1,OC192:EVT,WKSWPR-2,TC,01-10,08-12-00:\″Protection Switch, OC192
      port, Manual switch, MSS, 4F\″″
```

```
;
```

Path protection group

The following example retrieves an autonomous switch to protection for a 1+1 path protection group:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-1-25,STS1:EVT,WKSWPR-2,TC,01-10,08-12-00:\″Protection Switch, STS1
      CS, SF\″″
;
```

System clock

The following example retrieves a switch event for clock mode protection type:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″SYSTEM,SYST:EVT,HLDOVRSYNC,TC,11-09,12-00-00,,,:\″Protection switch,clock
      mode,Forced Holdover\″″
;
```

System timing reference

The following example retrieves a switch event for system timing reference protection type:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″EXTREF1,SYST:EVT,PS,TC,11-09,12-00-00,,,:\″Protection switch,sync
      reference,Manual switch\″″
;
```

Station timing reference

The following example retrieves a switch event for output timing reference protection type:

```
RTRV-LOG-PROTNSW:LT-UNITE::123456;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
```

```
   "LINE2,OUTT:EVT,INHSWPR,SC,11-09,12-00-00,,,:\"Protection switch,output
      reference,Lockout\""
;
```

Error responses

Validation rule: matching log entry

There is at least one matching log entry for the specified *startdate* and *starttime* if replytype equals DBSYNC.

Error code: `SNVS – Status, Not in Valid State, no matching log entry`

Related TL1 messages

- "REPT EVT: Report event" (p. 3-2121)
- "REPT SW: Report switch" (p. 3-2188)
- "RTRV-LOG-ALM: Retrieve log alarm" (p. 3-2454)
- "RTRV-LOG-NTFCN: Retrieve notification log" (p. 3-2464)

References

GR-833-CORE

RTRV-LOG-SECU: Retrieve log security

The **RTRV-LOG-SECU** command can be initiated by a user to retrieve the security information log activity from the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S5

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-LOG-SECU**:*tid*::*ctag::[startdate,starttime]*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **startdate** – Start Date
   The date for the starting point of the report. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set of the format YY-MM-DD
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
4. **starttime** – Start Time
   The time for the starting point of the report. If both *startdate* and *starttime* are omitted, the entire log is retrieved. Both *startdate* and *starttime* must be specified, or both must be omitted.
   Type: value set of the format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents the hours, MM the minutes, SS the seconds.
   Default: none

Output format

After receiving the **RTRV-LOG-SECU** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "login_id:umb:action:ccb:[uid_block]:[com_block]:[spec_block]:ccde"
```

```
      ...
   ″login_id:umb:action:ccb:[uid_block]:[com_block]:[spec_block]:ccde″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`login_id`** – Login ID
   The login identification of the user. It is the same as the user ID uid. See definition of *uid* in "ACT-USER: Activate user" (p. 3-24).
2. umb – Update Management Block
   The parameters within this block are positionally independent and are specified using a name-defined construct of parameter=value in a comma-separated list.
   - date
     The date of the user action.
     Type: value set of the format YY-MM-DD
     Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the last two digits of the year, MM the month, DD the day.
   - time
     The time of the user action
     Type: value set of the format HH-MM-SS
     Range: H, M, S are digits. Leading zeros are not suppressed. HH represents the hours, MM the minutes, SS the seconds.
3. **`action`** – Action
   This parameter indicates the reason for the log entry.
   Type: text string.
   Range:
   - For reasons for a logout entry following values are defined: ″inactivity timeout″ ″forced logout″ ″self logout″
     For reason for login failure following values are defined: ″expired passwd″ ″invalid uid″ ″invalid passwd″ ″locked out uid″
     For other reasons following values are defined: ″NE security variable change″
4. **`ccb`** – Command Code Block
   This parameter contains the name of the command.
   Type: Character string.
   Range: The names of the commands described in the Administration & Security except the RTRV -Commands.

5. **uid_block** – User ID Block
   This parameter contains the user ID of the user, who is affected by the logged action. If the action does not affect a specific user (e.g. modification of NE security parameters) the uid_block is empty.
   Type: Character string.
   Range: 1 – 10 character string or empty string.
6. com_block – Common Block
   It contains the common block of the command which is logged in the log. Please refer to the specifications of the commands. There exist only one exception: Passwords will be not logged.
7. spec_block – Specific Block
   It contains the specific block of the command which is logged in the log. Please refer to the specifications of the commands.
8. state_block – State Block
   It contains the state b lock of the command which is logged in the log. Please refer to the specifications of the commands.
   Note: The state block is not used in the commands to be logged. This parameter will be always empty.
9. **ccde** – Completion Code
   The completion code of the command.
   Type: Alphanumeric string
   Range: COMPLD or the 4-character error code.

Detailed behavior description

If the error log of the system does not contain any records at all or no records exist matching the specified startdate and starttime an empty reply is returned.

Example input/output

The following example shows the successful completion of **RTRV-LOG-SECU** command by the NE:

RTRV-LOG-SECU:LT-UNITE-123::123456;

```
   LT-UNITE-123 00-01-05 08:00:00
M  123456 COMPLD
   ″hugo:DATE=00-01-02,TIME=07-59-59:\″NE security variable change\″:ED-NE-
      SECU:::thrshld=30,intrvl=60,age=90:COMPLD″
   ″nbguser1:DATE=00-01-03,TIME=06-59-59:\″inactivity timeout\″:::::COMPLD″
   ″Admin1:DATE=00-01-04,TIME=09-59-59:\″invalid passwd\″:ACT-USER::::IDNV″
;
```

Related TL1 messages

- “ACT-USER: Activate user” (p. 3-24)
- “RTRV-LOG-ALM: Retrieve log alarm” (p. 3-2454)

- "RTRV-LOG-NTFCN: Retrieve notification log" (p. 3-2464)
- "RTRV-LOG-PROTNSW: Retrieve log protection switch" (p. 3-2468).

RTRV-LOPTF: Retrieve low-order path termination function

The **RTRV-PTF** command retrieves the attributes of a low-order path termination function (PTF) located on a TransMUX pack

VT1.5 LOPTF entities exist by changing the TransMUX capacity to 2G5. 28 LOVT instances will be created.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-LOPTF**-*modifier:tid:aid:ctag[:::spec_block];*

Input parameters

Command parameters

1. **modifier**
The modifier specifies the type of the low-order tributary.
Type: Alphanumeric value set.
Range: TMSVT.
Default: None.
2. **tid** – Target Identifier
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
The access identifier specifies a low-order tributary to be operated on.
Type: AID
Range: TMuxSLONumAID.
The following wildcard values are accepted:
   - **all**, **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID
   - **all** for Simple_TMuxSPTNumAID
   - **all** for Simple_TMuxSLONumAID

   Default: None.
4. **ctag** – Correlation Tag
Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **dyn** – Retrieve Dynamic Parameters
   his parameter controls whether dynamic data (output parameters) is reported with the command reply or not. It is only possible to retrieve those data if the aid input parameter does not address multiple entities at the same time (no wildcarding allowed). If omitted, the default value is assumed.
   The following output parameters are considered to be dynamic:
   - *ptrc*, *ptrcst*, *rsiglb*, *trailstate*

   Type: Alphanumeric value set.
   Range:
   - YES: Report dynamic parameters in the reply as well
   - NO: Do not report dynamic parameters in the reply.

Default: NO.

Output format

If the NE fully complies with the **RTRV-LOPTF** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid,aidtype[::spec_block]"
   <1 or more of the line above>
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: TMuxSLONumAID.
   The following values are not used:
   - **all**, **usall** for Simple_UnivSlotAID
   - **all** for Simple_UnivSlotAID

- **`all`** for Simple_TMuxSPTNumAID
- **`all`** for Simple_TMuxSLONumAID

No wildcards are allowed.
Default: None.

2. **`aidtype`** – Access Identifier Type
The *aidtype* specifies the type of the tributary.
Type: Alphanumeric value set.
Range: TMSVT.
Default: None.

Spec block parameters

1. **`degthr`** – Signal Degrade Threshold
This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range (as exponents of 10): –5; –6; –7; –8; –9, that means, the range is 10^{-9} to 10^{-5}.
Default: –6.

2. **`dexcthr`** – Excessive Degrade Threshold
This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (in contrast to **`RTRV-VCGTRIB`**, **`RTRV-VC3`**, **`RTRV-STS1`**).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range (as exponents of 10): –3; –4; –5, that means, the range is 10^{-5} to 10^{-3}.
Default: –3.

3. **`explptrc`** – Expected LO Path Trace Identifier
This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Hexadecimal number without a preceding “H”.
Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
Default: 89000000000000000000000000000000

4. **lopn** – LO ASA Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID. For *modifier* = TMSVT, an ASAP profile of type *PATHTERM* is required.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: 1 to 24 characters.
Default: DEFAULT.

5. **lptimais** – LO Trace Identifier Mismatch AIS Insertion
This parameter specifies whether a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
Type: Alphanumeric value set.
Range:
   - DISABLE
   - ENABLE

   Default: The default value depends on interface standard:
   - SONET: DISABLE.

6. **lptimmon** – LO Path Trace Identifier Mismatch Monitor
This parameter specifies the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte specific-string mode).
This parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
   - DISABLE
   - ENABLE

   Default: DISABLE

7. **ptrc** – Incoming Path Trace
This parameter reports the accepted low-order path trace identifier (J1 or J2). The length depends on the value of the parameter *ptrcrfmt*. The status of this value is contained in parameter *ptrcst*. If *ptrcst* indicates NORMAL, then *ptrc* contains a valid value. If *ptrcst* indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to not meeting the basic acceptance criteria or intra-NE communication problems, and the output parameter *ptrc* is omitted from the response.

This parameter is not supported for the following modifiers:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Hexadecimal

Range: the range depends on the parameter *ptrcrfmt* specified in **`RTRV-PTF`** output parameters:

- 2: hexadecimal digits if *ptrcrfmt* = 2
- 32: hexadecimal digits if *ptrcrfmt* = 16
- 128: hexadecimal digits if *ptrcrfmt* = 64.

Default: None.

8. **`ptrcrfmt`** – Path Trace Read Format
The parameter *ptrcrfmt* specifies the format of the path trace (J1 or J2 byte) to be received (sink).
The parameter is not supported for the following modifiers:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Numeric string.
   Range:
   - 2: for non-specific repeating byte
   - 16: for 16-byte specific string
   - 64: for 64-byte string.

   To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.
   Default: 2.
9. **`ptrcst`** – Status of Incoming Path Trace
This parameter shows the status of the value of the incoming Path Trace Identifier (J1 or J2) which is available with parameter *ptrc*.
The parameter is not supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - NORMAL, *ptrc* contains a valid value
   - UNAVAILABLE, no valid value for *ptrc* could be obtained by the system
   - AIS, no valid value for *ptrc* could be obtained due to SF condition

   Default: None.
10. **`ptrcwfmt`** – Path Trace Write Format
This parameter specifies the format of the path trace (J2 byte) to be sent (source).
Type: Numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16-byte specific string

Default: 2.

11. **`rsiglb`** – Received Signal Label
    This parameter reports the accepted trail signal label (C2 or V5). The parameter is not reported if the *trailst* parameter indicates a fail condition, that means, if it has the value AIS or UNAVAILABLE.
    The parameter is not supported for the following modifiers:
    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: 00, 01, …, FE, FF, that means two hexadecimal digits.
    Default: None.
12. **`tmonmd`** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.
    Type: Alphanumeric value set.
    Range:
    - MON
    - NMON

    Default: NMON.
13. **`trailst`** – Trail Status
    This parameter reports the accepted trail status value.
    Type: Alphanumeric value set.
    Range:
    - AIS for Alarm Indication Signal in case of Trail Signal Fail Condition
    - UNAVAILABLE for unavailable
    - NORMAL

    Default: None.
14. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J2) to be sent (source). For the 16-byte path trace format, the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    Type: Hexadecimal number without a preceding “H”.
    Range: The range depends on the parameter *ptrcwfmt* specified in **`RTRV-PTF`** output parameters
    - length: 32 hexadecimal digits.
    - value: specific string acc. ITU/ETSI mode 1.

    Default: 00.

Detailed behavior description

The **RTRV-LOPTF** does not generate a REPT DBCHG message. The command retrieves all port parameter as described in section “Output parameters”. It does not matter whether the entity is really in use or not.

Example input/output

Retrieval of all LO tributaries of a STS1 being substructured to 28 VT1.5:

```
RTRV-LOPTF-TMSVT:UNITE-NE:1-1-#-#-39-tms16-1-all:123456;

   UNITE-NE 08-03-01 11:11:00
M  123456 COMPLD
   ″1-1-#-#-39-tms16-1-1, TMSVT::\
    degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,\
    lopn=DEFAULT,lptimais=DISABLE,lptimmon=DISABLE,\
    ptrcrfmt=2,ptrcwfmt=2,tmonmd=NMON,trc=00″
   ″1-1-#-#-39-tms16-1-2, TMSVT::\
    degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,\
    lopn=DEFAULT,lptimais=DISABLE,lptimmon=DISABLE,\
    ptrcrfmt=2,ptrcwfmt=2,tmonmd=NMON,trc=00″

          <listing of more VT1.5 related LONIM’s of this HO Trib>
          <ending with>

   ″1-1-#-#-39-tms16-1-28, TMSVT::\
    degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,\
    lopn=DEFAULT,lptimais=DISABLE,lptimmon=DISABLE,\
    ptrcrfmt=2,ptrcwfmt=2,tmonmd=NMON,trc=00″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-LOPTF** command, too.

Validation rule: Check *aid* and *dyn* consistency

If the command is applied with an **aid** parameter containing wildcards (**all** at any position) and the **dyn** parameter is set to YES, then the command is denied.

Error code: `IDRG – Input, Data Range Error`

Example: Given the *aid* refers to 1-1-#-#-39-tms16-1-all and the *dyn* parameter is set to YES, then the command will be rejected.

Related TL1 messages

- "ED-LOPTF: Edit lower-order path termination function" (p. 3-427)
- "ED-LOPTF notification" (p. 3-432)
- "ENT-LOPTF notification" (p. 3-1606)
- "DLT-LOPTF notification " (p. 3-202)

RTRV-LONIM: Low-order non-intrusive monitor

The **`RTRV-LONIM`** command retrieves the attributes of a low-order non-intrusive monitor (LONIM, for example, OP***n***, EP155, or TransMUX).

The existence of the LONIM entities depends on the functionality:

LOXC

LONIM entities exist as long as a high-order tributary has been substructured. They are represented by entities only on lowest transmission rate with the maximum number of instances. The substructure depends on the interface standard of the related high-order tributary.

As result of a substructure creation low-order instances will be created.

The LONIM entities can be:

- SDH: 63 LOVC instances,
- SONET: 28 LOVT instances.

All of those entities can be always retrieved or modified although settings will only take effect for those entities which are indicated in the actual substructure.

TransMUX

DS1 LONIM entities exist as long as the TransMUX capacity of the approriate TransMUX card (where the DS1s are located) is unequal 0.

The LONIM entities can be:

- SONET: 28 DS1 instances

Example for LOXC

Assuming a VC4 (1st VC4 of port 1 in slot 1) is substructured to 3 VC3 (inputsig=33) since three unidirectional LOVC3 cross-connections have been established with that VC3s as input each. A RTRV-LONIM command with the input aid 1-1-#-#-1-1-1-all would deliver 63 responses although only 3 of them are in use (1-1-#-#-1-1-1-1, 1-1-#-#-1-1-1-22 and 1-1-#-#-1-1-1-43). The LONIM 1-1-#-#-1-1-1-2 can be modified without having any effect.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-LONIM-*modifier*:*tid*:*aid*:*ctag*::[:::*spec_block*],

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the low-order tributary. The purpose is to distinguish between: – SDH/SONET tributaries.
   Type: Alphanumeric value set.
   Range:
   - ALL
   - LOVC
   - LOVT
   - DS1

   The *modifier* value **ALL** does not include DS1 tributaries.
   The **ALL** modifier is only useful if AIDs with wildcards are used which have a bigger range than one single port or if the interface standard of the LONIM entities is unknown.
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   The access identifier specifies a low-order tributary to be operated on.
   Type: AID
   Range: GenLOTribAID, DS1NumAID.
   The following wildcard values are accepted:
   - *all*, *usall* for Simple_UnivSlotAID,
   - *all* for Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID
   - *all* for Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
   - *all* for Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID

   The motivation to allow a retrieval of all low-order tributaries at once (in combination with the ALL modifier) is that NE internally all slots and ports must be queried anyway to fill the response. If the retrieve command would be more restrictive then multiple commands would be necessary to retrieve all low-order tributaries which results internally to an unnecessary multiple full query.
4. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec Block Parameters

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic data (output parameters) is reported with the command reply or not. It is only possible to retrieve those data if the aid input parameter does not address multiple entities at the same time (no wildcarding allowed). If omitted the default value is assumed.
   The following output parameters are considered to be dynamic:
   - ptrc
   - ptrcst
   - rsiglb
   - trailst

   Type: Alphanumeric value set.
   Range:
   - YES: Report dynamic parameters in the reply as well
   - NO: Do not report dynamic parameters in the reply.

Output format

If the network element fully complies with the RTRV-LONIM request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid,aidtype[::spec_block]"
   <1 or more of the line above>
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the Output Parameter section for the **RTRV-HDR** command.

Output parameters

Command Parameters

1. **aid** – Access Identifier
   The access identifier specifies the low-order tributary whose attributes are reported.
   Type: AID
   Range: GenLOTribAID, DS1NumAID.
   The following values is not used:
   - *all*, *usall* for Simple_UnivSlotAID,
   - *all* for Simple_SPortAID, Simple_TMuxAPortAID, Simple_TMuxSPortAID

- *all* for Simple_PTNumAID, Simple_TMuxAPTNumAID, Simple_TMuxSPTNumAID
- *all* for Simple_LONumAID, Simple_DS1NumAID, Simple_TMuxSLONumAID

No wildcards are allowed.

2. **`aidtype`** – Access Identifier Type
   The *aidtype* specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: LOVC, LOVT, DS1

Spec Block Parameters

If a third of a high-order tributary has been substructured to a low-order VC-3 then only the first of the 21 referring LOVC entities will contain VC3 related contents – the other 20 entities will represent (currently unused) VC12 related contents. VC3 contents differs from VC12 contents e.g. in the range of the path trace receive format (VC3 allows 64 byte mode exclusively).

Configurations of unused entities is possible – even for performance monitoring. However, performance monitoring bins are not retrievable.

If a used entity changes its contents from VC3 to VC12 related contents or vice versa then all parameter for this entity will be set back to the referring default (note that the default values are transmission rate dependent) except for performance monitoring:

- Possibly enabled performance monitoring will not be stopped,
- SES threshold settings will be set back to default (since different)
- TCA profile assignment remains unchanged.

Other (VC12 related) entities of the same third within the referring high-order cross-connection (VC4) will not be changed at all.

1. **`crslpbkstat`** – Cross-connect Loopback Status
   This parameter shall report the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).

   The parameter is not supported for modifier:

   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.

Range:

- YES: for enabled loopback
- NO: for disabled loopback.

Default: NO.

2. **degthr** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. Note that only the POISSON mode is supported by HW for low-order tributaries. Therefore this parameter will be always valid (i.e. in contrast to RTRV-VCGTRIB, RTRV-VC3, RTRV-STS1).
   The parameter is not supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range: –5, –6, –7, –8, –9; i.e. range is 10^{-9} to 10^{-5}.
   Default: –6.
3. **dexcthr** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. Note that only the POISSON mode is supported by HW for low0order tributaries. Therefore this parameter will be always valid (i.e. in contrast to RTRV-VCGTRIB, RTRV-VC3, RTRV-STS1).
   The parameter is not supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Integer.
   Range: –3, –4, –5; i.e. range is 10^{-5} to 10^{-3}.
   Default: –3.
4. **explptrc** – Expected LO Path Trace Identifier
   This parameter specifies the Expected Low Order Path Trace Identifier (J1 or J2). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   The parameter is not supported for modifier:
   - DS1

   Option: If the parameter is not supported, then the parameter is omitted.
   Type: Hexadecimal number without a preceding “H”.
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1
   Default: 89000000000000000000000000000000
5. **feppm** – Far End Path PM Enable
   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.

6. **fmt** – Digital Signal Format for DS1
This parameter is used to specify the digital signal format expected to be received from the DS1 at the specified port(s). The TransMUX circuit pack supports DS1 path PM; it does not support DS1 line PM.
The parameter is only supported for modifier:
- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
- SF: Super Frame, supports near-end DS1 path and line PM only.
- ESF: Extended Super Frame, supports near-end and far-end DS1 path and line PM.

Default: ESF.

7. **lonimpn** – LO NIM ASAP Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
The ASAP type depends on the *modifier*:
- for LOVC/LOVT modifier, an ASAP of type PATH is required
- for DS1 modifier, an ASAP of type DS1 is required

Type: Alphanumeric string. A hyphen (“ – -”) is allowed in the name additionally.
Range: 1 to 24 characters
Default: DEFAULT.

8. **lptimmon** – LO Path Trace Identifier Mismatch Monitor
This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format ptrcrfmt is set to 16 (16-byte-specific-string mode).
The parameter is not supported for modifier:
- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE.
Default: DISABLE.

9. **neppm** – Near End Path PM Enable
This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE
Default: DISABLE.

10. **`outpm`** – Outgoing Near End Path PM Enable
The parameter *outpm* specifies whether the performance monitoring for the near end in outgoing direction is enabled or disabled for this tributary.

The parameter is only supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: DISABLE, ENABLE
Default: DISABLE.

11. **`pmlpfeses`** – PM LO Path Far End SES Threshold
This parameter specifies the Low Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end.
The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Integer.
Range:

The range depends on the tributary type in input direction, which can be derived from inputsig of the related PTF:

- 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
- 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
- 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary)

Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary).

12. **pmlpneses** – PM LO Path Near End SES Threshold
    This parameter specifies the Low Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end.
    The behavior depends on *modifier* and the transmission rate of this low-order tributary. For modifiers which include more than one transmission rate, the transmission rate can be derived from the parameter *inputsig* of the related PTF entity.

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Integer.
    Range:

    The range depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF:

    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a LO-VC3 tributary)
    - 1, 2, 3, …, 1998, 1999, 2000 (if the entity represents a VC12 tributary)
    - 1, 2, 3, …, 3998, 3999, 4000 (if the entity represents a VT1.5 tributary).
    - 1, 2, 3, …, 7998, 7999, 8000 (if the entity represents a DS1 tributary).

    Default: The default value depends on the tributary type in input direction, which can be derived from *inputsig* of the related PTF: 2400 (if the entity represents a LO-VC3 tributary); 600 (if the entity represents a VC12 or VT1.5 tributary); 1544 (if the entity represents a DS1 tributary).

13. **ptca** – Path TCA Profile Pointer
    The parameter *ptca* specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    The TCA Profile type depends on the modifier.

    - for LOVC *modifier*, a TCA Profile of type LOVC is required
    - for LOVT *modifier*, a TCA Profile of type LOPATH is required
    - for DS1 *modifier*, a TCA Profile of type DS1 is required

    Type: Alphanumeric string.
    Range: 1 … 24 characters.
    Default: DEFAULT0.

14. **ptrc** – Incoming Path Trace
    The parameter ptrc shall report the accepted low order path trace identifier (J1 or J2). The length depends on the value of the parameter ptrcrfmt. The status of this value is contained in parameter ptrcst. If ptrcst indicates NORMAL, then ptrc contains a valid value. If ptrcst indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to not meeting the basic acceptance criteria or intra-NE communication problems, and the output parameter ptrc is omitted from the response.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Hexadecimal

Range: the range depends on the parameter ptrcrfmt specified in RTRV-LONIM Output Parameters

- 2, hexadecimal digits if *ptrcrfmt* = 2
- 32, hexadecimal digits if *ptrcrfmt* = 16
- 128, hexadecimal digits if *ptrcrfmt* = 64.

15. **`ptrcrfmt`** – Path Trace Read Format
This parameter specifies the format of the path trace (J1 or J2 byte) to be received (sink).

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Numeric string.

Range:

- 2 (default), for non-specific repeating byte
- 16, for 16 byte specific string
- 64, for 64 byte string (only applicable if the entity represents a LO-VC3 tributary).

To check whether the entity represents a LO-VC3 tributary or not, the parameter *inputsig* of the related PTF entity must be checked.

16. **`ptrcst`** – Status of Incoming Path Trace
This parameter shows the status of the value of the incoming Path Trace Identifier (J1 or J2) which is available with parameter *ptrc*.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- NORMAL, *ptrc* contains a valid value
- UNAVAILABLE, no valid value for *ptrc* could be obtained by the system
- AIS, no valid value for *ptrc* could be obtained due to SF condition

Default: None.

17. **`rsiglb`** – Received Signal Label
This parameter shall report the accepted trail signal label (C2 or V5). The parameter is not reported if the trailst parameter indicates a fail condition, i.e. has the value AIS or UNAVAILABLE.

The parameter is not supported for modifier:

- DS1

Option: If the parameter is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: 00, 01, …, FE, FF, i.e. two hexadecimal digits.
Default: None.

18. **`tcamd`** – TCA mode
    This parameter is only available for SDH mode and will therefore only be reported if the output parameter *aidtype* is LOVC. The parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable for SDH:

    - TR-only. (also known as the implicit reset method)
    - TR-RTR. (also known as the explicit reset method).

    For SONET, the parameter *tcamd* is not supported. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

    The parameter is not supported for modifier:

    - DS1
    - LOVT

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: TR, TRRTR

    Default:

    - SDH mode: TRRTR
    - SONET mode: TR (internal).

    For SONET the parameter tcamd does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.

19. **`tmonmd`** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.
    Type: Alphanumeric value set.
    Range: MON, NMON.
    Default: NMON.

20. **`trailst`** – Trail Status
    This parameter shall report the accepted trail status value.

    The parameter is not supported for modifier:

    - DS1

    Option: If the parameter is not supported, then the parameter is omitted.
    Type: Alphanumeric string

Range:

- AIS for Alarm Indication Signal in case of Trail Signal Fail Condition
- UNAVAILABLE for unavailable
- NORMAL.

Default: None.

Detailed behavior description

The RTRV-LONIM does not generate a REPT DBCHG message. The command shall retrieve all low order tributary parameter related to specified AID as described in section Output Parameters. It does not matter whether the entity is really in use or not.

No LO NIMs exist at the HO side of a conversion cross-connection. Therefore RTRV-LONIM does not return any values related to the HO side of a conversion cross-connection.

Example input/output

Retrieval of all LO tributaries of a VC4 being substructured to 3 VC3:

```
RTRV-LONIM-LOVC:LT-UNITE:1-1-#-#-39-16-1-all:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-39-16-1-1,LOVC::crslpbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=2400,pmlpneses=2400,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON″
   ″1-1-#-#-39-16-1-2,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON″
          <listing of more unused VC12’s ending with>
   ″1-1-#-#-39-16-1-21,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON″
   ″1-1-#-#-39-16-1-22,LOVC::crslpbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=2400,pmlpneses=2400,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON″
   ″1-1-#-#-39-16-1-23,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
```

```
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
          <listing of more unused VC12's ending with>
   "1-1-#-#-39-16-1-42,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
   "1-1-#-#-39-16-1-43,LOVC::crslpbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=2400,pmlpneses=2400,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
   "1-1-#-#-39-16-1-44,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
          <listing of more unused VC12's ending with>
   "1-1-#-#-39-16-1-63,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
;
```

Retrieval of all LO tributaries of a VC4 being substructured to 63 VC12:

```
RTRV-LONIM-LOVC:LT-UNITE:1-1-#-#-39-16-1-all:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   "1-1-#-#-39-16-1-1,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"
   "1-1-#-#-39-16-1-2,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
    tcamd=TRRTR,tmonmd=NMON"

          <listing of more VC12 related LONIM's of this HO Trib>
          <ending with>

   "1-1-#-#-39-16-1-63,LOVC::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,ptrcrfmt=2,\
```

```
    tcamd=TRRTR,tmonmd=NMON"
;
```

Retrieval of all LO tributaries of an STS1 being substructured to 28 VT1.5:

```
RTRV-LONIM-LOVT:LT-UNITE:1-1-#-#-39-16-1-all:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   "1-1-#-#-39-16-1-1,LOVT::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,\
    neppm=DISABLE,pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,\
    ptrcrfmt=2,tmonmd=NMON"
   "1-1-#-#-39-16-1-2,LOVT::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,\
    neppm=DISABLE,pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,\
    ptrcrfmt=2,tmonmd=NMON"

          <listing of more VT1.5 related LONIM's of this HO Trib>
          <ending with>

   "1-1-#-#-39-16-1-28,LOVT::crslpbkstat=NO,degthr=-6,\
    dexcthr=-3,explptrc=89000000000000000000000000000000,\
    feppm=DISABLE,lonimpn=DEFAULT,lptimmon=DISABLE,\
    neppm=DISABLE,pmlpfeses=600,pmlpneses=600,ptca=DEFAULT0,\
    ptrcrfmt=2,tmonmd=NMON"
;
```

Retrieval of a single VC3 with dynamic parameters:

```
RTRV-LONIM-LOVC:LT-UNITE:1-1-#-#-39-16-1-43:123456::dyn=YES;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   "1-1-#-#-39-16-1-43,LOVC::crslpbkstat=NO,degthr=-6,dexcthr=-3,\
    explptrc=89000000000000000000000000000000,feppm=DISABLE,\
    lonimpn=DEFAULT,lptimmon=DISABLE,neppm=DISABLE,\
    pmlpfeses=2400,pmlpneses=2400,ptca=DEFAULT0,\
    prtc=89000000000000000000000000000000,ptrcst=NORMAL,\
    ptrcrfmt=2,rsiglb=00,tcamd=TRRTR,tmonmd=NMON,trailst=NORMAL"
;
```

Retrieval of single DS1 tributary:

```
RTRV-LONIM-DS1:UNITE-NE:1-1-#-#-15-tma1-1-d28:123456;

   UNITE-NE 08-03-01 11:11:00
M  123456 COMPLD
   "1-1-#-#-15-tma1-1-d28,DS1::fmt=SF,lonimpn=DEFAULT,\
```

```
   outpm=ENABLE,pmlpneses=1544,ptca=DEFAULT0,tmonmd=NMON"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-LONIM** command, too.

Validation rule: Check LONIM::modifier against PTF::ptfstd

If the command is applied with a modifier which does not match to the interface standard parameter *ptfstd* of the owning PTF entity, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

The following combinations are allowed (*modifier istd*):

- LOVC, SDH
- LOVT, SONET
- DS1, SONET
- ALL, SDH or SONET

Validation rule: Check *aid* and *dyn* consistency

If the command is applied with an *aid* parameter containing wildcards (all at any position) and the *dyn* parameter is set to YES, then the command is denied.

Error code: `IDRG – Input, Data Range Error`

Example: Given the aid refers to 1-1-#-#-39-16-1-all and the dyn parameter is set to YES then the command will be rejected.

Related TL1 messages

- "ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)

RTRV-LOXCPM: Retrieve performance monitoring data of LOXC pack

The **`RTRV-LOXCPM`** command is initiated by a user to request the network element to send the current and/or historical Performance Monitoring (PM) data associated with performance monitoring points located on an LOXC pack. For each value in the *montype* parameter of the **`RTRV-PM`** input Parameters section, PM counter values will be stored in the current 15-minute period, the 32 previous 15-minute periods (past 8 hours worth of data), current day, and previous day periods.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

`RTRV-LOXCPM`-*modifier*:*tid*:*aid*:*ctag*::[*montype*], [*monlev*], [*locn*],,*tmper*, [*mondat*], [*montm*];

Input parameters

Command parameters

1. **`modifier`**
   This parameter defines the type of performance monitoring point.
   Type: Alphanumeric value set.
   Range: The following value is for SONET only: LOVT1.
   The following values are for SDH only: LOVC3, LOVC12.
2. **`tid`** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **`aid`** – Access Identifier
   Type: AID
   Range: GenLOTribAID
   Wildcard values in parts of both AIDs are usable as follows:
   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_SPortAID,
   - *all* in Simple_PTNumAID,
   - *all* in Simple_LONumAID.
   - *all* in Simple_TMuxSLONumAID.

   Default: None.

4. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
5. **montype** – Monitored parameter type
   This is the monitored parameter type for which data is requested. If the value is omitted, this parameter defaults to ALL.
   Type: value set.
   Range:
   - The following values are for LOVT1 modifiers only: CVP, ESP, SESP, UASP, FC-P, CV-PFE, ES-PFE, SES-PFE, UAS-PFE, FC-PFE
   - The following values are for LOVC3 and LOVC12 modifiers only: VC-N-ES, VC-N-SES, VC-N-BBE, VC-N-UAS, VC-F-ES, VC-F-SES, VC-F-BBE, VC-F-UAS

   The following value is also allowed for all input modifiers: ALL.
   Default: ALL.
6. **monlev** – Monitored Level
   This parameter provides the level for the monitored PM parameter. The parameter is only evaluated if the value of *montype* parameter is ALL.
   Type: value set.
   Range:
   - 1-UP: Means the NE will report all non-zero data and all data which is marked as non-valid (parameter validity).
   - 0-UP: Means the NE will report zero and all positive data.

   Default: 1-UP.
7. **locn** – Location
   Location for which the PM data is monitored. The parameter value is only considered if the *montype* value is ALL; otherwise it is ignored for the calculation of the command reply (i.e has no impact on the command).
   Type: value set.
   Range:
   - NEND: Near-End PM data is requested.
   - FEND: Far-End PM data is requested.
   - ALL: All near-end and far-end data is requested. It applies only to input direction.

   Default: ALL.
8. **tmper** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.

9. **`mondat`** – Monitored Date

   This requests the beginning date of the interval for which the PM data is to be reported. For both values of tmper parameter, the maximum allowable range of valid dates includes the current and previous one day.
   Type: value set.

   Range:

   - MM-DD: Month-Day. A leading zero (first M or first D) is optional and may be omitted.
   - ALL: Current and previous day.

   Default: Current date

10. **`montm`** – Monitored Time

    This parameter specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. The user is able to specify a monitored time in three modes:

    a. A monitored time interval (i.e., from hour-minutes to hour-minutes: HOD-MOH&&THOD-TMOH).

    b. A single time interval (i.e., the hour and minutes for the appropriate 15 minute data: HOD-MOH).

    c. All applicable beginning times for the given *mondat* value (i.e., all).

    Where HOD-MOH represents the start of the requested PM retrieval period (hours and minutes) and THOD-TMOH represents the end of the requested PM retrieval period. The value ALL must be used if the *mondat* value provided is ALL.
    Type: Character string.

    Range:

    - HOD-MOH: This specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. This parameter may be omitted or have one of the following values: Hour Of Day- Minute Of Hour, where HOD ranges from 0 to 23 and MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.
    - THOD-TMOH: To Hour Of Day-To Minute Of Hour, where THOD ranges from 0 to 23 and TMOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0. Parameter grouping is used when specifying THOD-TMOH. For example, 08-15&&09-00 specifies the HOD-MOH and THOD-TMOH of 08-15 and 09-00, respectively. THOD-TMOH of $$-$$ indicates the current Hour Of Day-Minute Of Hour. THOD-TMOH of ##- ## indicates the previous 15-min interval.
    - ALL: All applicable beginning times for the given *mondat* value.

    If the *tmper* value is 1-DAY, the *montm* parameter will not be used.
    Default:
    If THOD-TMOH is not specified, only the PM data for the retrieval period specified by HOD-MOH will be retrieved.
    If HOD-MOH is omitted, the PM period for the current time will be reported.

Output format

If the NE fully complies with the RTRV-LOXCPM request, the following normal completion response is returned:

```
    sid date time
M   ctag COMPLD
   "aid:montype,monval,[validity],[tca],[locn],,tmper,[mondat],[montm]"
   . . . . .
   . . . . .
   . . . . .
;
```

Note: The trailing comma has to be present even if the optional parameter is not supported for a dedicated release.

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the Output Parameter section for the **RTRV-HDR** command.

Normal completion message – empty reply

If there is no PM data stored within the NE, then the following response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

1. **aid** – Access Identifier
   Description: Access identifier.
   Type: AID
   Range: GenLOTribAID
   Wildcard values (e.g. *all* or *usall*) in any parts of both AIDs is not used.
2. **montype** – Monitored Parameter Type
   See montype in the input parameters section.
3. **monval** – Monitored Value
   This parameter contains the measured value of the parameter specified in montype.
   Type: Integer.
   Range: 0-4294967295
   The specified range is the maximal range taking all PM parameters into consideration. Specific PM parameters may have a smaller range.

   W.r.t. value range for the specific PM parameters see

   - "Output parameters" section in "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982) (if thresholding is applicable for that parameter)
   - PM parameter introduction section (if thresholding is not applicable for that parameter).

4. **`validity`** – validity indicator
   This parameter indicates the validity of the PM data.
   Type: Integer.
   Range: 1: valid; (omitted): non-valid.
   If the value is omitted, it indicates that the validity is ″non valid″.
5. **`tca`** – Threshold Crossing Alert
   Type: value set.
   Range:
   For the current interval: *tca* indicates:
   - in TCA mode TR, that the current value exceeds the provisioned threshold
   - in TCA mode TR-RTR, that a *tca* occurred since the beginning of this interval or that the *tca* reset condition was not fulfilled in any previous interval since the last tca occurred.

   If the value is omitted, it indicates
   - in TCA mode TR, that the current value is below the provisioned threshold
   - in TCA mode TR-RTR, that no tca occurred since the beginning of this interval and that the *tca* reset condition was fulfilled at least in one previous interval.

   For previous intervals, the *tca* parameter indicates the state at the end of the interval.
6. **`locn`** – Location
   Location for which the PM data is monitored.
   Type: value set.
   Range:
   - NEND: Near-End PM data is requested
   - FEND: Far-End PM data is requested.
7. **`tmper`** – Time Period
   See in the input parameters section.
8. **`mondat`** – Monitored Date
   This parameter specifies the beginning date of the interval for which the PM data is reported. The parameter can be omitted in case of current date.
   Type: value set.
   Range: MM-DD: Month-Day
   Default: Current date
9. **`montm`** – Monitored Time
   This parameter specifies the beginning time of day for the reported pm register. The parameter is optional and may be omitted for the reporting of current bins (registers).
   Type: Character string.
   Range: HOD-MOH: HOD ranges from 0 to 23; MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.

Detailed behavior description

This parameter shall

- only return those parameters, which are specified by the input parameters
- omit parts of the response (maybe to return an empty response) if related performance monitoring has not been enabled (independent whether this occurs due to a specific or wildcard query)
- omit parts of the response due to a wildcard query (maybe to return an empty response) if the related entity does not exist
- omit an entity within the response if the entity is not in use for the specified transmission rate or if the entity is currently not in use at all (LOVC12 only)
- return an empty response for non-existing LONIM entities.

Examples:

If a command is executed for a specific existing LOVT1 entity to retrieve all parameters but the entity has only near-end (thus no far-end) performance monitoring enabled, then a response will be returned containing only the near-end values.

If a command is executed for all LOVC3 entities of a specific VC4 PTF but this PTF is substructured into 1 VC3 and 42 VC12 then a response will be returned containing only the parameters for the one existing VC3.

If a command is executed for a specific LOVC12 entity but the related PTF is substructure into 3 VC3 signals then an empty response will be returned.

Further examples leading to empty responses:

- A command with a specific AID containing a GenLOTribAID for a DS3 port
- A command with a specific AID and modifier LOVC3 containing a Simple_LONumAID different than 1, 22, 43
- A command with a specific AID and modifier LOVC12 containing a Simple_LONumAID equal to 1, 22 or 43 and the related third of the VC4 substructure indicates a VC3 signal rate
- A command with a specific AID and modifier LOVT1 containing a Simple_LONumAID greater than 28.

Example input/output

The following example shows the successful completion of a **RTRV-LOXCPM** command by the network element for a LOVT1:

```
RTRV-LOXCPM-LOVT1:UNITE-NE:1-1-#-#-39-16-3-28:123456::\
vc-f-es,1-up,,,15-min ,,00-00&&08-00;

   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
   ”1-1-#-#-39-16-3-28:vc-f-es,17,,tca,,,15-min,,00-00”
```

```
   ″1-1-#-#-39-16-3-28:vc-f-es,10,,tca,,,15-min,,00-15″
   . . .
   ″1-1-#-#-39-16-3-28:vc-f-es,15,,tca,,,15-min,,08-00″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-LOXCPM** command. See also subsection 'Partial Completion on RTRV Commands' w.r.t. handling of internal error (for example, communication to XCn fails) for a subset of replies.

Related TL1 messages

- "RTRV-EPM: Ethernet performance monitoring" (p. 3-2352)
- "DLT-TCA-PROF: Delete TCA profile" (p. 3-259)
- "ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
- "ENT-TCA-PROF: Enter TCA Profile" (p. 3-1894)
- "RTRV-TCA-ASGNMT: Retrieve TCA assignment" (p. 3-2979)
- "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982)

References

GR-833-CORE

RTRV-LPBK (cross-connection): Retrieve loopback (cross-connection)

The **RTRV-LPBK**-*modifier* command can be initiated by a user to retrieve information on cross-connection loopback(s) in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-LPBK**-*modifier*:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the rate of the currently active cross-connection loopback for the specified tributary on which the **RTRV-LPBK** command acts on.
   For the adaptive rate case this means that the *modifier* does not specify the rate (constituent signal type) of the tributary, but the cross-connection loopback rate.
   Type: Alphanumeric value set.
   Range:
   - ALL
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - LOVC3

- LOVC12
- LOVT1.

2. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
3. **`aid`** – Access Identifier
   This parameter specifies the tributary for which loopback information is requested to be retrieved.
   Type: GenTribAID or GenLOTribAID.
   Range: The **`all`**-parameter is accepted for the slot-, port-, high-order trib- and optionally for the low-order-trib-part of the AID. In addition, the value **`vall`** is accepted for the high-order trib-part and the value **`usall`** is accepted for the slot-part of the AID. If the slot-/port-/high-order trib-part is set to **`all`** or **`usall`** then all high-order loopbacks are retrieved but those referring to GE***n***, ES64U, and OPT***n***. To get those, the **`vall`** value must be used at high-order trib-part of the AID. If the low-order-trib-part is set to **`all`** then all low-order loopbacks of the system are retrieved. The value **`vall`** for the high-order trib-part of the AID is not allowed in this case.
   For example the following *aids* are valid:
   - 1-1-#-#-all-all-all specifying all HO tribs for all ports in the whole shelf
   - 1-1-#-#-2-1-all specifying all HO tribs of the first port of the pack in slot-number=02
   - 1-1-#-#-3-vall-all specifying all HO tribs of all ports of the (GE***n***, ES64U, or OPT***n***) pack in slot-number=03
   - 1-1-#-#-all-all-all-all specifying all LO tribs for all HO tribs and ports in the whole shelf

   However the following example shows an invalid aid: 1-1-#-#-all-vall-all-all there is no low order tributary on GE***n*** or OPT***n*** packs:
   1-1-#-#-all-vall-all-all: there is no lower-order tributary on GE***n***, ES64U, or OPT***n*** packs
4. **`ctag`** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

Normal completion message

In response to a valid **`RTRV-LPBK-`***modifier* command, the following output is returned:

```
   sid date time
M  ctag COMPLD
   "aid:rate:spec_block"
       ...
       ...
   "aid:rate:spec_block"
```

;

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the tributary for which loopback information is retrieved.
   Type: GenTribAID or GenLOTribAID.
   Range: No **`all`** or **`usall`** or **`vall`** value shell be possible for any part of the aid..
2. **`rate`**
   This parameter indicates the cross-connection rate. For a detailed description please refer to parameter *rate* in “RTRV-CRS: Retrieve cross-connection” (p. 3-2306).
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
   - LOVC3
   - LOVC12
   - LOVT1.

Spec block parameters

1. **`crslpbkstat`** – Cross-connect Loopback Status
   This parameter specifies the current cross-connection loopback status of the considered tributary *aid*.
   The value *YES* will be used in case the tributary is loop-backed.
   Note that the tributary does not need to be involved in a cross-connection – but can

be in loopback also without cross-connection.
Note that no dedicated value is specified for the case of a non loop-backed tributary since such a tributary will no be in output.
Type: Alphanumeric value set.
Range: YES.

Detailed behavior description

The **RTRV-LPBK** command allows the retrieval of loopback information for tributaries, regardless if they are involved in cross-connections or not. Tributaries which are not involved in a cross-connection loopback are not present in the reply output.

Interface Standard Independence

No command validation between port interface-standard and modifier will be performed. The output parameter *rate*, however, is filled with the rate according to the interface standard of the port.

Example Input/output

Example:

The example shows the loopback status for the 13th trib of the pack in slot 07, which has an existing loopback:

```
RTRV-LPBK-VC3:UNITE-NE:1-1-#-#-7-1-13:123456

   sid date time
M  ctag COMPLD
   "1-1-#-#-7-1-13:STS1:crslpbkstate=YES"
;
```

Related TL1 messages

- “OPR-LPBK: Operate cross-connection loopback” (p. 3-2035)
- “RLS-LPBK: Release loopback (facility)” (p. 3-2198)

References

GR-199-CORE

RTRV-LPBK (facility): Retrieve loopback (facility)

The **RTRV-LPBK**-*modifier* command can be initiated by a user to retrieve information on facility loopback(s) in the NE. Facility herein is identical to port – either an optical port of an OP***n*** pack or the optical port of a GE***n*** pack – no other ports (like timing ports, UserByte ports etc) are settable to loopback.

As long as there is loopback active on a port, the *abnormal* LED lights.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-LPBK**-*modifier:tid:aid:ctag;*

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the port rate on which the **RTRV-LPBK** command acts on. (In principle this parameter is redundant, because the port-rate is implicitly given by the port-AID (*SPortAID*), but in order to align the facility loopback command with the tributary loopback command this parameter is retained.)
   Type: Alphanumeric value set.
   Range:
   - ALL
   - STM1E
   - STM1
   - STM4
   - STM16
   - STM16T
   - STM64
   - STM256
   - EC1
   - OC3
   - OC12
   - OC48
   - OC48T

- OC192
- OC768
- 1GE
- 1GEPL
- 10GEPL
- T3

2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   Access Identifier of the port for which the loopback state is returned.
   Type: AID
   Range: SPortAID (when using OP***n***, EP155, OPT***n*** packs or EC1 interfaces on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_SPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is accepted.

   APortAID (when using DS3 interfcaes on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_APortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is accepted.

   EPortAID (when using GE***n*** packs) as defined in AID structure. The values:
   - all for Simple_EPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is accepted.
4. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Output format

In response to a valid RTRV-LPBK command, the following output is returned:

```
   sid date time
M ctag COMPLD
   "aid:rate:spec_block"
       ...
       ...
   "aid:rate:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier of the port for which the loopback state is returned.
   Type: AID
   Range: SPortAID (when using OP***n***, EP155, OPT***n*** packs or EC1 interfaces on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_SPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   APortAID (when using DS3 interfcaes on the EP51 pack) as defined in AID structure. The values:
   - all for Simple_APortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.

   EPortAID (when using GE***n*** packs) as defined in AID structure. The values:
   - all for Simple_EPortAID, usall for Simple_UnivSlotAID, all for Simple_UnivSlotAID, all for Simple_ShelfAID and all for BayAID is excluded.
2. **rate**
   This parameter indicates the port *rate*.
   Type: Alphanumeric value set.
   Range:
   - STM1E
   - STM1
   - STM4
   - STM16
   - STM16T
   - STM64
   - STM256
   - EC1
   - OC3
   - OC12
   - OC48
   - OC48T
   - OC192
   - OC768

- 1GE
- 1GEPL
- 10GEPL
- T3.

Spec block parameters

1. **lpbk** – Loopback
   This parameter specifies the current near-side loopback status of the output *aid*. A near-side loopback means that the signal on the incoming port is looped to the outgoing port.
   If *lpbk* = YES then the port is (near-side) loop-backed.
   If *lpbk* = NO then there is no (near-side) loopback existing for the port.
   Note: At most one loopback (*lpbk* or *fsflpbkstat*) can be set to YES for the same port at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
2. **fsflpbkstat** – Far-Side-Facility-Loopback-Status
   This parameter specifies the current far-side facility loopback status of the output *aid*. A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port.
   If *fsflpbkstat* = YES then the port is (far-side) loop-backed. If *fsflpbkstat* = NO then there is no (far-side) loopback existing for the port.
   Note: At most one loopback (*lpbk* or *fsflpbkstat*) can be set to YES for the same port at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).

Detailed behavior description

The **RTRV-LPBK** command allows the retrieval of near-side and far-side loopback information for ports.

If no port exists with the specified interface standard, rate, and type (*modifier*) at the AID or in the AID range, then the command shall complete and return an empty response.

For the GE***n*** the lpbk parameter is always set to NO due to the fact that the GE (EPort) only allows the far-side loopback possibility.

Example input/output

Suppose the port 1 of the pack in slot 7 has a existing near-side loopback:

```
RTRV-LPBK-ALL:UNITE-NE:1-1-#-#-7-1:123456

   sid date time
M  ctag COMPLD
   "1-1-#-#-7-1:OC48:lpbk=YES,fsflpbkstat=NO"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-LPBK** command.

Related TL1 messages

- "OPR-LPBK: Operate facility loopback" (p. 3-2044)
- "RLS-LPBK: Release loopback (facility)" (p. 3-2198)

References

GR-199-CORE

RTRV-MANROUTE-LAN: Retrieve manual route at LAN port

The **RTRV-MANROUTE-LAN** command is used to retrieve manual route entries at LAN ports, which are used as UNI control channels.

Properties

Priority: S1

Abortable: Yes

Input format

Command syntax: **RTRV-MANROUTE-LAN**:*tid:[aid]:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   Default: If this field is omitted, the report will display the manual route entries of all lan ports assigned to the UNI node.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*.

Output format

If the **RTRV-MANROUTE-LAN** is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:spec_block"
       ...
   "aid:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the "RTRV-HDR: Retrieve header" (p. 3-2417) command.

Output parameters

Command parameters

1. **aid** – Target Identifier
   This parameter indicates the LAN port AID. The value *lanall* is not allowed.
   Type: AID.
   Range: LanPort AID.

Spec block parameters

1. **ip_ad** – destination IP Address
   This parameter is used o indicate the destination IP address to which a manual route is created. This is the LAN i/f address of UNI-C
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255. The values 0.0.0.0 and 1.1.1.1 are not allowed.
2. **ip_ad_uni** – Node ID of UNI-C
   This parameter is usedo indicate the router ID of the UNI-C to which a manual route is created.
   Type: Four dot separated numeric strings
   Range: Each numeric string: 0 .. 255.
3. **osinode**
   This parameter is used to indicate the OSI node assigned to the LAN port.
   Type: AID.
   Range: uninode.

Detailed behavior description

The command shall retrieve the manual route entries for LAN ports which are used for UNI control channel, i.e. are assigned to UNI node. If no manual route entries exist, the command shall return an empty list. The initial value of the manual routes it an empty list.

Example input/output

The following example shows the result of RTRV-MANROUTE-LAN:

```
RTRV-MANROUTE-LAN:NE-UNITE:1-1-lan2:123456;
   NE-UNITE 06-06-30 11:01:00
M  123456 COMPLD
   ″1-1-lan2:ip_ad=135.248.206.72,ip_ad=135.248.143.102,ip_
      snm=255.255.0.0,osinode=uninode″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **`RTRV-MANROUTE-LAN`** command.

Validation rule: Not allowed to use LAN1 for UNI

If LAN1 is specified in the AID, then the command is denied.

Error code: `IDNV – Input, Data Not Valid, LAN1 not allowed for UNI`

Validation rule: LAN port not used for UNI

If the LANport specified in the AID is not in use for UNI, i.e. is not assigned to the uninode, then the command is denied.

Error code: `SNVS – Status, Not in Valid State, LAN port not in use for UNI`

Related TL1 messages

- "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610)
- "DLT-MANROUTE-LAN: Delete manual route at LAN port" (p. 3-203)

RTRV-MANROUTE-MCN: Retrieve manual IP route for management communication

The **RTRV-MANROUTE-MCN** command is used to retrieve the provisioned manual MCN IP routes from a gateway NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-MANROUTE-MCN**:*tid:[aid]:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter indicates the LAN port AID.
   Type: AID.
   Range: LanPort AID.
   The keyword "lanall" is allowed for Simple_LANPortAID.
   Default: Omitting this field has the same effect as the keyword "lanall", that means, the report will display the manual MCN route entries of all possible LAN ports assigned to the MCN.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **RTRV-MANROUTE-MCN** request is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″aid:spec_block″
       .
       .
     <one line for each Manual MCN IP Route entry>
       .
   ″aid:spec_block″
```

```
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Target Identifier
   This parameter indicates the LAN port AID. The value *lanall* is not allowed.
   Type: AID.
   Range: LanPort AID.

Spec block parameters

1. **dest_ipad** – Destination IP Address
   The IP address of an IP node which can be reached via one of the LAN ports.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: None.
2. **dest_ipmask** – Destination IP Subnet Mask
   The IP subnet mask of the IP node which can be reached via the LAN port.
   Type: Four dot separated numeric strings
   Range: Each numeric string: 0 … 255.
   Default: None.
3. **nexthop_ipad** – Next Hop IP Address
   The IP address of the next hop routing node via which the Destination IP Address can be reached.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 … 255. The value 0.0.0.0 is used if the IP node defined by *dest_ipad* is connected to the LAN port directly. The values 1.1.1.1 and 127.0.0.1 are not allowed.
   Default: None.

Detailed behavior description

The command retrieves all provisioned manual MCN IP routes. The output of the command is not sorted.

Example input/output

The following example shows the result of **RTRV-MANROUTE-MCN**:

```
RTRV-MANROUTE-MCN:UNITE-NE::123456;
```

```
   NE-UNITE 08-06-20 11:01:00
M  123456 COMPLD
   "1-1-lan2:dest_ipad=192.168.100.72,dest_ipmask=255.255.0.0,nexthop_
      ipad=192.168.1.83"
   "1-1-lan2:dest_ipad=192.168.200.165,dest_ipmask=255.255.0.0,nexthop_
      ipad=192.168.1.83"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-MANROUTE-MCN** command.

Related TL1 messages

- "ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
- "ENT-MANROUTE-MCN notification" (p. 3-1618)
- "DLT-MANROUTE-MCN: Delete one manual MCN IP route entry" (p. 3-207)
- "DLT-MANROUTE-MCN notification" (p. 3-209)

RTRV-MAP-NEIGHBOR: Retrieve map neighbor

The **RTRV-MAP-NEIGHBOR** command will retrieve a list of all immediate OSI neighbors to the local NE (node or NE). This can include other NEs, as well as neighbors on the LAN, such as the CIT.

A neighbor entry will not appear in the report for channels allocated to the SCN_node_AID or the UNI_node_AID or the ENNI_node_AID.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-MAP-NEIGHBOR**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: TID
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: CTAG

Output format

If the **RTRV-MAP-NEIGHBOR** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   ″[rate]:thru[,lpaid]:spec_block″
   ...
   ″[rate]:thru[,lpaid]:spec_block″
;
```

Ordering of output lines: Output is sorted in ascending order first by *lpaid*.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **rate** – DCC Port Rate
   This parameter is only available for a DCC neighbor and provides the optical port rate as used in RTRV-*rr*.
   Type: Value set.
   Range:
   - STM1
   - STM4
   - STM16
   - STM64
   - STM256
   - OC3
   - OC12
   - OC48
   - OC192
   - OC768.

   Default: n/a – as determined by the port type.
2. **thru** – Connected Through
   This parameter specifies the type of connectivity between the NE and its neighbor.
   Type: value set.
   Range: DCC, LAN.
   Default: n/a – as determined by the type of connection.
3. **lpaid** – Local Port AID
   This parameter provides the SPortAID of the optical port in case thru=DCC or the LanPortAID in case thru=LAN.
   Type: AID.
   Range: SPortAID in case thru=DCC or LanPortAID in case thru=LAN as defined in section AID Structure.

Spec block parameters

1. **nsap** – Network Service Access Point
   This filed defines the ASAP of the neighbor.
   Type: 20 byte octet string in 40 digit hex representation.
   Range: The NSAP address/NET is composed of the following components:
   - l3idp (l3afi and l3idi)
   - l3dfi,

- l3org,
- l3res,
- l3rd,
- l3area,
- l3sys,
- l3se.l

See “RTRV-ULSDCC-L3: Retrieve layer 3 related parameters” (p. 3-3049) for details.

2. **`npaid`** – Neighbor Port AID
   This parameter is only available for DCC neighbors and provides the port AID of the optical port of the OSI neighbor. (Peer of *lpaid*)
   Since the neighbor NE might have a different AID structure it is interpreted as a string.
   Type: AID string.
   Range: up to 20 characters-
3. **`tid`** – Neighbor TID
   This parameter is only available for DCC neighbors. It provides the TID value of the neighbor NE.
   Type: TID
   Range: Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for legal values and a general description of TIDs.
4. **`dcctype`** – DCC Type
   This parameter is only available for DCC neighbors. It specifies whether the values displayed in the *spec_block* refer to the MS/LINE or RS/SECTION DCC.
   Type: value set.
   Range:
   - SECTION
   - LINE

   Default: n/a – depends on DCC channel type for which *spec_block* parameters are reported.

Detailed behavior description

The command retrieves per NE information about all neighbors which are physically connected to the local NE via DCC or LAN and about details of the connection as specified in the output command and *spec_block* parameters.

If there is a link failure between the local NE and its neighboring NE, or if the DCC is disabled, the neighbor NE entry will not appear in the report.

The command will not retrieve the neighborship between the internal osinodes of the NE. This is only relevant for the network level design and RTRV-MAP-NETWORK can be used for it.

A neighbor node entry shall not appear in the report for DCC channels allocated to the CN_node_AID or the UNI_node_AID. A neighbor node entry shall not appear in the report for LANs allocated to the SCN_node_AID or the UNI_node_AID or the ENNI_node_AID.

This command does not generate a **REPT DBCHG** notification.

Example input/output

The following example shows the result of a RTRV-MAP-NEIGHBOR command for an NE which has all DCCs (RS/MS) enabled for the port on an OP40 and the CIT LAN:

```
RTRV-MAP-NEIGHBOR:LT-UNITE::123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=SECTION″
   ″STM256:DCC,1-1-#-#-1-1:
nsap=39840F80000000000000000000000000000a400,npaid=1-1-#-#-1-1,tid=LT-
      UNITE2,dcctype=LINE″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-MAP-NEIGHBOR** command.

Related TL1 messages

- “ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack” (p. 3-1312)
- “RTRV-ULSDCC-L3: Retrieve layer 3 related parameters” (p. 3-3049)
- “RTRV-MAP-NETWORK: Retrieve network map” (p. 3-2526)

RTRV-MAP-NETWORK: Retrieve network map

The **RTRV-MAP-NETWORK** command will retrieve a list of OSI nodes, reachable from the local NE (NE). This can include other NEs, other nodes on the local NE, as well as neighbors on the LAN, such as the CIT.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-MAP-NETWORK**:*tid*:*aid*:*ctag*[::*level2*];

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: TID.
2. **aid** – Access Identifier
   This parameter specifies the osinode for which the OSI nodes is retrieved. The value *osinodeall* is not allowed.
   Type: AID.
   Range: OSI_node_AID | SCN_node_AID. The value *osinodeall* is not allowed.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: CTAG.
4. **level2** – IS Routing Level
   If set to N(o), report will list only the nodes in the local level 1 area of this node.
   If set to Y(es), report will list the nodes in the level 2 sub-domain (when *aid* specifies a level 2 node).
   If set to N(o) the list will include level-1-only and level-1+2 nodes.
   Note: For nodes where level 2 routing is not supported, the output will be empty when *level2* = Y.
   Note: The previous note implies that level2 is set to N, if info for osinode=scnnode is requested.
   Type: value set.

Range:

- N (default)
- Y.

Output format

If the RTRV-MAP-NETWORK request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   ″spec_block″
   ...
   ″spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Spec block parameters

1. **`level1`** – level 1 Router
   This parameter specifies whether reachable node is level 1 router.
   Type: Value set.
   Range:
   - Y (es)
   - N (o).
2. **`level2`** – Level 2 Router
   This parameter specifies whether reachable node is level 2 router.
   Type: Value set.
   Range:
   - Y (es)
   - N (o).
3. **`ip_ad1`** – IP Address 1
   This parameter is used to retrieve the IPv4 address of a reachable SCN node or LAN interface in the network map. This output parameter is only available when input aid is of type SCN_node_AID.
   Note: The IP address can be a host address or one of up to two LAN interfaces addresses. There is no order or semantics (i.e. whether IP address belongs to SCN

node or LAN interface) specified for ip_ad1, ip_ad2, ip_ad3.
Type: Four dot separated numeric strings or empty string.
Range: Each numeric string: 0 .. 255 or empty string.
Note: The empty string is reported when the IP address can not be retrieved. In output direction it is represented as \"\".

4. **`ip_ad2`** – IP Address 2
   This parameter is used to retrieve the IPv4 address of a reachable SCN node or LAN interface in the network map. This output parameter is only available when input aid is of type SCN_node_AID.
   Note: The IP address can be a host address or one of up to two LAN interfaces addresses. There is no order or semantics (i.e. whether IP address belongs to SCN node or LAN interface) specified for ip_ad1, ip_ad2, ip_ad3.
   Type: Four dot separated numeric strings or empty string.
   Range: Each numeric string: 0 .. 255 or empty string.
   Note: The empty string is reported when the IP address can not be retrieved. In output direction it is represented as as \"\".
5. **`ip_ad3`** – IP Address 3
   This parameter is used to retrieve the IPv4 address of a reachable SCN node or LAN interface in the network map. This output parameter is only available when input aid is of type SCN_node_AID.
   Note: The IP address can be a host address or one of up to two LAN interfaces addresses. There is no order or semantics (i.e. whether IP address belongs to SCN node or LAN interface) specified for ip_ad1, ip_ad2, ip_ad3.
   Type: Four dot separated numeric strings or empty string.
   Range: Each numeric string: 0 .. 255 or empty string.
   Note: The empty string is reported when the IP address can not be retrieved. In output direction it is represented as \"\".
6. **`ip_sid`** – SID
   This parameter is used to display the SID of a reachable SCN node in the network map. This output parameter is only available when input aid is of type SCN_node_AID.
   Note: Since the SID (MAC address) is unique it might be used to detect misconfigurations of duplicate IP addresses.
   Type: Hexadecimal number.
   Range: 000000000000 – FFFFFFFFFFFF.
7. **`net`** – Network Entity Title
   This field is the Network Entity Title (NET) of one reachable OSI node. The NET is the same as the NSAP with the last byte set to 00 (hex).
   This output parameter is only available when input aid is of type OSI_node_AID.
   Type: 20 byte octet string in 40 digit hex representation.
   Range: The NET is composed of the following components:
   - l3idp (l3afi and l3idi)
   - l3dfi,

- l3org,
- l3res,
- l3rd,
- l3area,
- l3sys,
- l3sel = 00 (hex).

See “RTRV-ULSDCC-L3: Retrieve layer 3 related parameters” (p. 3-3049) for details.

8. **tid** – Target Identifier
This parameter specifies the TID of the NE. This parameter is reported for entries of the local and remote NEs if OSI node 1 is specified as AID in the command. For OSI nodes 2 – 8 and the SCN node, the parameter is only reported for entries of the local NE.
Type: TID
Range:
   - Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for legal values of TIDs
   - ″?″ when the TID cannot be determined.

Detailed behavior description

The **RTRV-MAP-NETWORK** command retrieves per node a list of OSI or SCN nodes (depending on the input aid), reachable from the local network element (NE) via the OSI or SCN stack(s). This can include other network elements, other nodes of the local NE, as well as other systems on the LAN, such as the CIT.

In contrast to RTRV-MAP-NEIGHBOR or RTRV-SCN-NEIGHBOR, which retrieve info about physical neighbors and properties of the physical connection, this command shall retrieve info with respect to (I)IS-IS routing.

If the node specified by aid is provisioned as level 2 and level2=Y, the report shall list the level 2 nodes across the level 2 subdomain and the nodes in the local level 1 area of the selected node. The output parameters level1 and level2 contain the correct level 1, 2 indication.

If level2=N, the report shall list only the nodes in the local level 1 area of the selected node (including the default level 2 node and other level 2 nodes in the local level 1 area). The output parameters level1 and level2 are meaningless.

The local OSI system is also listed in the report.

OSI node not in created state

If the specified OSI node is not in created state, then the system shall return no response.

This command does not generate a **REPT DBCHG** notification.

Example input/output

The following example shows the result of a RTRV-MAP-NETWORK command for an input aid of type OSI_node_AID:

```
RTRV-MAP-NETWORK:LT-UNITE:osinode1:123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″tid=LT-UNITE-A,
   net=39840F80000000000000000000000601D0d56e000,
   ip_ad1=\″\″,ip_ad2=\″\″,ip_ad3=\″\″,level1=Y,level2=N″
   ″tid=?,net=39840F80000000000000000000000108334F1E600,
   ip_ad1=\″\″,ip_ad2=\″\″,ip_ad3=\″\″,level1=Y,level2=N″
   ″tid=?,net=39840F8000000000000000000000A0D103189900,
   ip_ad1=\″\″,ip_ad2=\″\″,ip_ad3=\″\″,level1=Y,level2=N″
   ″tid=LT-UNITE-B,
   net=39840F80000000000000000000000601D0D52D000,
   ip_ad1=\″\″,ip_ad2=\″\″,ip_ad3=\″\″,level1=Y,level2=N″
   ″tid=LT-UNITE-C,
   net=39840F80000000000000000000000601D0D562000,
   ip_ad1=\″\″,ip_ad2=\″\″,ip_ad3=\″\″,level1=Y,level2=N″

;
```

The following example shows the result of a RTRV-MAP-NETWORK command for an input aid of type SCN_node_AID:

```
RTRV-MAP-NETWORK:LT-UNITE:scnnode:123456;
   LT-UNITE 00-11-01 15:00:00
M  123456 COMPLD
   ″tid=LT-UNITE,ip_ad1=192.168.200.1,ip_sid=123456789012,level1=Y,level2=N″
   ″ip_ad1=192.168.200.8,ip_ad2=\″\″,ip_ad3=\″\″,ip_
      sid=123456789012,level1=Y,level2=N″
   ″ip_ad1=192.168.200.17,ip_ad2=\″\″,ip_ad3=\″\″,ip_
      sid=123456789123,level1=Y,level2=N″
   ″ip_ad1=192.168.200.23,ip_ad2=192.168.200.26,ip_ad3=192.168.200.29,ip_
      sid=123456789576,level1=Y,level2=N″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-MAP-NETWORK** command.

Validation rule: level2 parameter must not be present when *osinode* = scnnode

If the osinode parameter for which the network map should be retrieved is the *scnnode* the input parameter level2 is not allowed.

Error code:

`IDNC- Input, Data Not Consistent, level2 not allowed for scnnode`

Related TL1 messages

- "ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
- "RTRV-ULSDCC-L3: Retrieve layer 3 related parameters" (p. 3-3049)
- "RTRV-MAP-NEIGHBOR: Retrieve map neighbor" (p. 3-2522)
- "RTRV-SCN-NEIGHBOR: Retrieve SCN neighbor" (p. 3-2826)

RTRV-MAP-RING: Retrieve map ring

The **RTRV-MAP-RING** command retrieves the ring map for one MS-SPRing/BLSR protection group.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-MAP-RING**:*tid*:[*aid*]:*ctag*::[*rid*],*[mapsrc]*;

Important! To make sure that the complete ring map information will be given the **RTRV-RING-MAP** command should always be issued against the UNITE NE within the ring.

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **ctag** – Correlation Tag
   See definition in input parameters section "RTRV-HDR: Retrieve header" (p. 3-2417).

4. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.

5. **`mapsrc`** – Map source
   This parameter specifies, how the ring map is derived. The value DCCEI is used if the ring map is derived via the DCF function. The value NVMRING is used if the ring map is derived from non-volatile memory as used by the MS-SPRing/BLSR protection group. The value MANUAL is used, if the ring map data which was manually populated by the user and stored in RAM is shown.
   Note: If NVMRING is specified and there is a valid and complete ringmap in NVM, it will be reported. Otherwise, nothing will be reported.
   Type: Alphanumeric value set.
   Range: DCCEI, NVMRING, MANUAL.
   Default: DCCEI.

Output format

After receiving the **`RTRV-MAP-RING`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:rid,localnid:spec_block"
    ...
    ...
   "aid:rid,localnid:spec_block"
;
```

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the aid of 2F-/4F MS-SPRing/BLSR protection.
   Type: AID
   Range: TwoFibre_ProtGrpAID or FourFibre_ProtGrpAID
2. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
3. **`localnid`** – Local Node ID
   This parameter specifies the local node ID of the NE receiving the retrieve request (i.e. the node which has been addressed within RTRV-MAP-Ring command) as used by the MS-SPRing/BLSR APS protocol.
   Type: Numeric string.
   Range: 0 .. 15

Spec block parameters

1. **`nid`** – Node ID
   This parameter specifies the node ID of the NE within the ring as used by the MS-SPRing/BLSR APS protocol. It shall correspond to the first TID in the *linkid* block.
   Type: Numeric string.
   Range: 0 .. 15
2. **`linkId`** – Link Identifier
   This parameter specifies a link connecting two NEs within the ring. It has the following format:
   linkid=\"{*nename*,[*port*],*direction*},{*nename*,[*port*],*direction*}\"
   The first value set of *nename*, *port*, *direction* corresponds to the NE as indicated by parameter *nid*.
   The second value set of *nename*, *port, direction* corresponds to the NE reachable via the link.
   The port and direction may be omitted, if the value *mapsrc* is MANUAL.
   Type: inner string
   Range: see parameter description for *nename*, *port, direction* below.
   Note: The *linkID* as used in the ring map will refer to the ports of that link which is used for the automatic detection of the ring topology (i.e. to protection ports for asymmetric spans and to worker ports for symmetric spans).

3. **`nename`** – NE Name
   See definition of parameter *tid* "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **`port`** – port AID
   This parameter specifies the port being a member within the 2-/4-fibre MS-SPRing/BLSR protection group.
   Type: AID, in case *nename* denotes the local NE; AID string, in case nename denotes another NE in the ring.
   Range: SPortAID, in case of an AID; up to 20 characters otherwise
   Option: The parameter may be omitted if unknown due to link failure.
5. **`direction`** – Direction
   This parameter specifies the side role of the corresponding port within the protection group. The value E is used for the east side. The value W is used for the west side.
   Type: Alphanumeric value set.
   Range: E, W
   Option: The parameter may be omitted if unknown due to link failure.
6. **`continuity`** – Link Continuity Status
   This parameter specifies the link continuity status of a port. The following values are supported:
   - UP – The link is working.
   - DOWN – The link has failed.
   - OPEN – The link is non-existent, as in an open ring.
   - UNKNOWN – The link ID has not been established.
   - ISOLATED – The link is in an isolated part of a segmented ring.

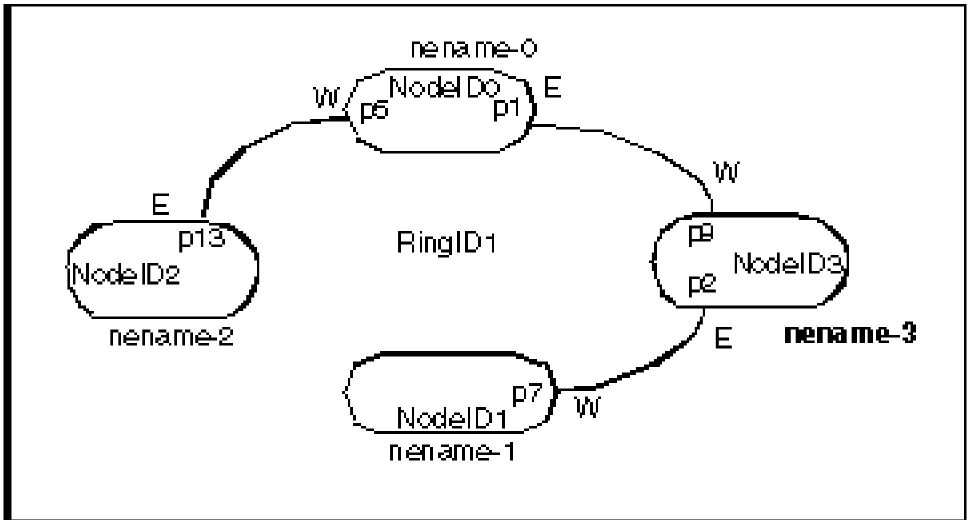
   Type: Alphanumeric value set.
   Range: UP, DOWN, OPEN, UNKNOWN, ISOLATED.
7. **`net`** – Network Entity Title
   This parameter specifies the NSAP (Network Service Access Point) with the last byte (SEL byte – see NSAP format) set to 00 (hex) for the node with TID as specified for the first *nename* within *linkid* list.
   Type: Hexadecimal digit string
   Range: 40 characters

Detailed behavior description

For open rings, the output starts at the end with the lower *nid* value and continuous to the other end. For closed rings, the output starts at the node with the lowest *nid* value and continues in the direction of the first node's east neighbor.

Example input/output

Ring map for an open ring – example 1

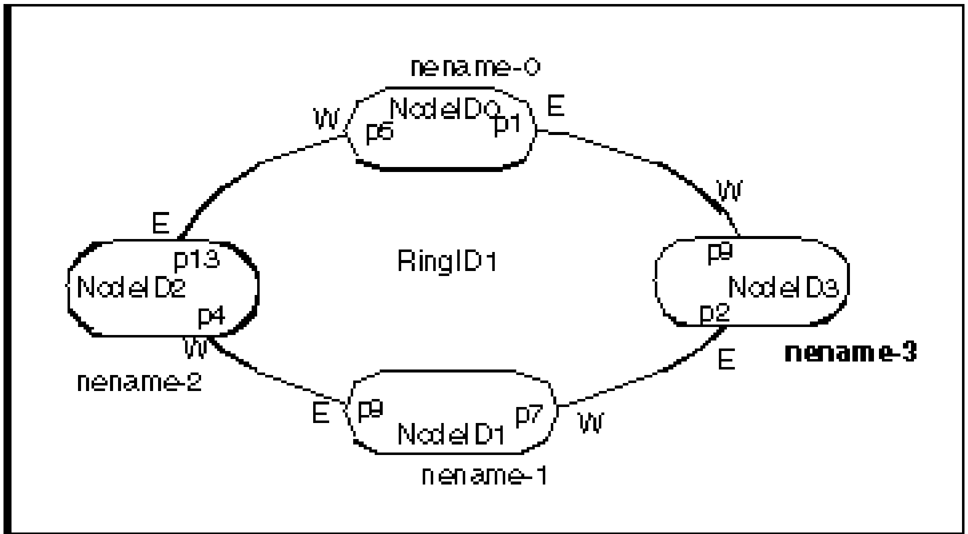


Open Ring (RingID1)

RTRV-MAP-RING for example 1:

```
RTRV-MAP-RING:NENAME-1:1-1-f01:123456;
   NENAME-1 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-f01:\″RingID1\″,1:nid=1,linkid=\″{nename-1,1-1-#-#-4-7,W},
   {nename-3,1-1-#-#-2-2,E}\″,continuity=UP,
net=39840F80000000000000000000000000a100″
   ″1-1-f01:\″RingID1\″,1:nid=3,linkid=\″{nename-3,1-1-#-#-3-9,W},
{nename-0,1-1-#-#-1-1,E}\″,continuity=UP,
net=39840F80000000000000000000000000a300″
   ″1-1-f01:\″RingID1\″,1:nid=0,linkid=\″{nename-0,1-2-#-#-2-5,W},
{nename-2,1-2-#-#-1-13,E}\″,continuity=UP,
net=39840F80000000000000000000000000a000″
   ″1-1-f01:\″RingID1\″,1:nid=2,linkid=\″{nename-2,,},
{nename-1,,}\″,continuity=OPEN,
net=39840F80000000000000000000000000a200″
;
```

Ring map for an closed ring – example 2



Ringmap for Closed Ring at nename-3

RTRV-MAP-RING for example 2:

```
RTRV-MAP-RING:NENAME-3::123456:\"RingID1\";
   NENAME-3 98-01-01 08:00:
M  123456 COMPLD
   "1-1-f01:\"RingID1\",3:NID=0,linkid=\"{nename-0,1-1-#-#-4-1,E},
{nename-3,1-1-#-#-2-9,W}\",continuity=UP,
net=39840F800000000000000000000000000a000"
   "1-1-f01:\"RingID1\",3:NID=3,linkid=\"{nename-3,1-1-#-#-3-2,E},
{nename-1,1-1-#-#-1-7,W}\",continuity=UP,
net=39840F800000000000000000000000000a300"
   "1-1-f01:\"RingID1\",3:NID=1,linkid=\"{nename-1,1-1-#-#-4-9,E},
{nename-2,1-1-#-#-3-1,W}\",continuity=UP,
net=39840F800000000000000000000000000a100"
   "1-1-f01:\"RingID1\",3:NID=2,linkid=\"{nename-2,1-1-#-#-2-13,E},
{nename-0,1-1-#-#-1-1,W}\",continuity=UP,
net=39840F800000000000000000000000000a200"
;
```

RTRV-MAP-RING for example 3

```
RTRV-MAP-RING:NENAME-3::123456:\"RingID1\",MANUAL;
   NENAME-3 98-01-01 08:00:
M  123456 COMPLD
   "1-1-f01:\"RingID1\",3:NID=0,linkid=\"{nename-0,,},
{nename-3,1-1-f01-ww-tr2-1,W}\",continuity=UNKNOWN"
   "1-1-f01:\"RingID1\",3:NID=3,linkid=\"{nename-3,1-1-f01-ew-tr3-1,E},
{nename-1,,}\",continuity= UNKNOWN"
   "1-1-f01:\"RingID1\",3:NID=1,linkid=\"{nename-1,,},
```

```
{nename-2,,}\",continuity=UNKNOWN"
   "1-1-f01:\"RingID1\",3:NID=2,linkid=\"{nename-2,,},
{nename-0,,}\",continuity=UNKNOWN"
;
```

Error responses

Validation rule: consistent aid and rid

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Related TL1 messages

- "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
- "RTRV-MAP-NEIGHBOR: Retrieve map neighbor" (p. 3-2522)
- "RTRV-MAP-NETWORK: Retrieve network map" (p. 3-2526)
- "ED-MAP-RING: Edit map ring" (p. 3-436)
- "SET-MAP-RING: Set map ring" (p. 3-3142)
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)

References

GR-199-CORE

RTRV-NADR: Retrieve ONNS network (domain) address data repository

Overview

The **RTRV-NADR** command requests information of the ONNS network (domain) address data repository, which contains address information of the own domain about TNA type, TNA address, logical port IDs for UNI and EDGE ports and associated client nodes for UNI ports.

The **RTRV-NADR** command operates on the current, active-memory-resident, ONNS data. The NVM data from NNDM is used during initialization to recover the ONNS local data repository. Data from other ONNS nodes is acquired by the flooding and exchange of LSAs.

Retrieving ALL Network address data repository (NADR) records

When the *dataType* = ALL, Network Navigator shall respond to a request to retrieve its Network Address Data Repository (NADR) by sending a complete report of the contents of the data repository specified. The request shall cause the ONNS node to send all information in the specified data repository. When the Network address data repository is retrieved, each TL1 response message contains a record with all of the data relevant to the address information part of the LSA.

Retrieving Network address data repository (NADR) records for any ASTN node ID

The command **RTRV-NADR** with *dataType* = LSAID and *componentID* = ASTNNodeID shall retrieve the LSA address records that relates to the specified node with the ASTNNodeID. This ASTNNodeID can be the address of any node in the ONNS domain.

Retrieving Network address data repository (NADR) records for a specific port class

The command **RTRV-NADR** with *dataType* = portClass and *componentID* = UNI or EDGE shall retrieve the LSA address records that relates to the specified portClass.

REPT PTHCHG or REPT NNCHG is not generated

The **RTRV-NADR** command shall not generate a REPT PTHCHG or REPT NNCHG notification.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: NA

Abortable: No

Input format

The **RTRV-NADR** command has an input parameter *dataType* that defines what type of information is expected in the *componentID* field that determines what portion of the data repository records is retrieved.

Command syntax: **RTRV-NADR**:*srcTid:srcAid:ctag::dataType[,componentID]*;

Input parameters

Command parameters

1. **srcTid**
   Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique Tid used by Host NE. See also the definition of tid in RTRV-HDR.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The srcTid parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier *srcAid* of the ONNS controller OS command interface port.
   Type: AID.
   Range: snn
   The keyword **all** is allowed up to shelf level.
3. **ctag**
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
4. **dataType**
   This parameter specifies, which part of the address information is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.
   Range: ALL – retrieves the entire (NADR) Network address data repository in RAM; LSAID – retrieves the LSA address record related to the ASTN node ID specified in the *componentID* must be an *ASTNnodeID*, PORTUSAGE – retrieves the LSA address records related to the port class specified in the *componentID* must be UNI or EDGE.
5. **componentID**
   componentID depends on the previous *dataType* parameter and is one of:
   not represented, when *datatype* = ALL
   nodeID, when *datatype* = LSAID (any ASTN node ID)
   portClass, when *datatype* = PORTUSAGE
   nodeID – Node Identifier:
   Identifier of an ASTN node.

Type: Four dot separated numeric strings.
Range: 0..255, for each numeric string.

portClass

Ports used by ONNS with additional address information.
Type: Alphanumeric value set.

Range:

- EDGE ONNS client port, where the signal enters/leaves the ONNS domain, unmanaged by ONNS, but used as the source or destination port of an ONNS connection
- UNI ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client, used as the source or destination port of an ONNS connection

Output format

The Network address data repository (NADR) record structure

The complete NADR of one node consists of *LSAAddrRecords* for all known nodes (including the reporting node itself) in the topology. Within a *LSAAddrRecord* all individual TNARecords with the associated logical port IDs are listed.

Network address data repository (NADR) without data

An addressed node without data in the NADR shall response only the HeaderRecord with *NumberOfLSAs* = 0. The network address data repository (NADR) Record is composed of an NADR_HeaderRecord and multiple *LSAAddrRecords* which are described below:

- **NADR_Record ::=**
  <NADR_HeaderRecord>
  <LSAAddrRecord_l>
  …
  <LSAAddrRecord_m>
- **NADR_HeaderRecord ::=**
  ″nadrID,dataType[=componentID],srcNodeASTNNodeID, NumberOfLSAs=LSA_Count″
- **LSAAddrRecord ::=**
  <LSAAddr_HeaderRecord>
  <TNARecord_1>
  <LogicalPortIDList_1>
  …
  <TNARecord_n>
  <LigicalPoerList_n>
- **LSAAddr_HeaderRecord ::=**
  ″LSAAdvertisedRouter=origNodeASTNNodeID,TNACount=TNAcount, portclass″

- **TNARecord ::=**
  ″TNAtype=TNAaddress[,ClientNodeID=nodeID]″
- **LogicalPortIDList ::=**
  ″LogicalPortIDs=lpid_1[&lpid_2&…&lpid_5]″
  [″LogicalPortIDs=lpid_6&lpid_7&…&lpid_10″]
  …
  [″LogicalPortIDs=lpid_n-4&lpid_n-3&…&lpid_n″]
  There is not more then 5 logical port IDs per line to avoid a line break.

Output format with normal data retrieve completion report for *dataType* = ALL

If the RTRV-NADR request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″NADR_HeaderRecord″
   ″LSAAddr_HeaderRecord1″
     ″TNARecord_1-1″
       ″LogicalPortIDList_1-1″
     ″TNARecord_1-2″
       ″LogicalPortIDList_1-2″
   ...
   ″LSAAddr_HeaderRecord4″
     ″TNARecord_4-1″
       ″LogicalPortIDList_4-1″
     ″TNARecord_4-2″
       ″LogicalPortIDList_4-2″
     ″TNARecord_4-3″
       ″LogicalPortIDList_4-3″
>
   srcTid date time
M  ctag COMPLD
   ″LSAAddr_HeaderRecord5″
     ″TNARecord_5-1″
       ″LogicalPortIDList_5-1″
     ″TNARecord_5-2″
       ″LogicalPortIDList_5-2″
     ″TNARecord_5-3″
       ″LogicalPortIDList_5-3″
     ″TNARecord_5-4″
       ″LogicalPortIDList_5-4″
   ...
   ″LSAAddr_HeaderRecord9″
     ″TNARecord_9-1″
       ″LogicalPortIDList_9-1″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completionCode* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

NADR_HeaderRecord

1. **`nadrID`**
   Indicator for the responded data repository
   Type: Alphanumeric value set.
   Range: NADR, Network address data repository in RAM.
2. **`dataType`**
   This parameter specifies, which part of the address information is to be retrieved. Furtherore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.

   Range:

   - ALL: retrieves the entire (NADR) Network address data repository in RAM.
   - LSAID: retrieves the LSA address records related to the ASTN *nodeID* specified in the *componentID* must be an *ASTNnodeID*
   - PORTUSAGE: retrieves the LSA address records related to the port class specified in the *componentID* must be UNI or EDGE
3. **`componentID`**
   This parameter depends on the previous dataType parameter and is one of: not represented (when *datatype* = ALL), nodeID (when datatype = LSAIP (any node ID), portClass (when *datatype* = PORTUSAGE).
   `nodeID` – Node Identifier
   Identifier of an ASTN node.
   Type: Four dot separated numeric strings.
   Range: for each numeric string: 0..255
   `portClass`
   Ports used by ONNS with additional address information.
   Type: Alphanumeric value set.

   Range:

   - EDGE ONNS client port, where the signal enters/leaves the ONNS domain, unmanaged by ONNS, but used as the source or destination port of an ONNS connection
   - UNI ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client, used as the source or destination port of an ONNS connection

   Note: If *dataType* = ALL, *componentID* and the “=” sign are not reported.
4. **`srcNodeASTNNodeID`** – Reporting Node ID
   This parameter is the ASTN node ID of the ONNS node replying to this command.
   Values: Four dot separated numeric strings, 0 – 255 for each numeric string.

5. **`LSA_Count`**
   This parameter indicates the number of LSA address records in the report.
   Type: Integer
   Range:
   - >0: number of LSAs with address records
   - 0: no EDGE or UNI port(s) with additional address information are defined at this node or within the network

   In case of LSA_Count = 0 only the *NADR_HeaderRecord* is displayed in the output.

LSAAddrRecord

For each reported node of the topology one LSAAddrRecord per reported port class are reported.

1. LSAAddrHeaderRecord
   All parameters in the LSAAddrHeaderRecord describe properties of the *LSAAdvertisedRouter*.
   a. **`origNodeASTNNodeID`**
      The ASTN node ID of the advertiser of the LSA with the address data following the *LSAAddrHeaderRecord*.
      Type: Four dot separated numeric strings.
      Range: 0 – 255 for each numeric string.
   b. **`TNACount`**
      Number of reported TNAs of *LSAAdvertisedRouter* are provisioned on this node.
      Range: >0
   c. **`portClass`**
      Ports used by ONNS with additional address information.
      Type: Alphanumeric value set.
      Range:
      – EDGE ONNS client port, where the signal enters/leaves the ONNS domain, unmanaged by ONNS, but used as the source or destination port of an ONNS connection
      – UNI ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client, used as the source or destination port of an ONNS connection

TNARecord

For each TNA within the LSAAddrRecord one TNARecord is reported.

1. **`TNAType`**
   This parameter specifies the address type of the TNA address.
   Type: Alphanumeric value set.

Range:

- IPV4TNAADDR (default), TNA address is a IPv4 address
- IPV6TNAADDR, TNA address is a IPv6 address
- NSAPTNAADDR, TNA address is a NSAP address

2. **TNAAddress**
   TNAAddress depends on the previous TNAtype parameter and is one of:
   - IPv4TNAAddress, when TNAtype=IPV4TNAADDR
     Identifier of a TNA address in IPv4 format.
     Type: Four dot separated numeric strings.
     Range: 0 … 255, for each numeric string.
   - IPv6TNAAddress, when TNAtype=IPV6TNAADDR
     Identifier of a TNA address in IPv6 format.
     Type: Eight hyphen separated hexadecimal numbers.
     Range: 0 … FFFF, for each hexadecimal number.
   - NSAPTNAAddress, when TNAtype=NSAPTNAADDR
     Identifier of a TNA address in NSAP format.
     Type: Alphanumeric value set (20byte)
     Range: up to 20 alphanumeric strings with hexadecimal values 00..FF.
3. **nodeID** – Client Node ID
   IP address of the UNI client derived from the TNAAddress. This parameter is optional and only available for ports of portClass UNI.
   Type: Four dot separated numeric strings. Example: 135.13.44.61
   Range: 0 … 255, for each numeric string.
4. **LogicalPortIDList**
   The list of *LogicalPortIDs* is a list of numeric values indicating specific physical ports within a TNAaddress. The *logicalPortID* values are shown in the form: lpid1([&lpid2& …&lpidx]).
   Type: &-separated list of numeric strings.
   Range: lpid1([&lpidX])* where X <= 512 where lpidn = 0 … 42946967295. The value 512 for X results as theoretical value from the number of slots multiplied by number of possible ports per slot in 1675 LambdaUnite MSS.

Example input/output

Example NADR output for the **RTRV-NADR** command with ALL

The **RTRV-NADR** command with *dataType* = ALL produces this example output result:

RTRV-NADR:LT-ONNS-61::123456::ALL;

```
   LT-ONNS-61 2002-07-14 09:45:28
M  123456 RTRV
   "NADR,ALL,135.13.44.61,NumberOfLSAs=4"
   "LSAAdvertisedRouter=135.13.44.61,TNACount=4,UNI”
     “IPV4TNAADDR=135.13.3.119,ClientNodeID=135.13.44.61”
       “LogicalPortIDs=1&234&33678”
     “IPV6TNAADDR=1ED-2FC2-3-F365-E290-119-45-A321,
ClientNodeID=135.13.3.113”
       “LogicalPortIDs=3459&23&3134”
      “IPV6TNAADDR=3ED-2FC2-3-F365-F290-119-45-A360,
ClientNodeID=135.13.3.113”
       “LogicalPortIDs=345&234&367”
     “IPV4TNAADDR=128.52.765.119,ClientNodeID=135.13.3.21”
       “LogicalPortIDs=145&678&3342&230598&3210"
>
   LT-ONNS-61   2002-01-14   09:45:29
M  123456 RTRV
       “LogicalPortIDs=2319”
   "LSAAdvertisedRouter=135.13.3.21,TNACount=3,UNI"
     “IPV4TNAADDR=112.45.163.113,ClientNodeID=135.13.144.221”
       “LogicalPortIDs=132”
     “IPV4TNAADDR=133.16.3.145,ClientNodeID=135.13.14.216”
       “LogicalPortIDs=459&342901”
     “IPV4TNAADDR=139.133.32.119,ClientNodeID=135.13.3.23”
       “LogicalPortIDs=340918&3298&4590&3210&4509&3190"
>    ....
   LT-ONNS-61   2002-01-14   09:45:30
M  123456 COMPLD
   "LSAAdvertisedRouter=135.13.3.113,TNACount=3,UNI"
     “IPV4TNAADDR=135.16.3.120,ClientNodeID=135.13.3.114”
       “LogicalPortIDs=320”
     “IPV4TNAADDR=176.113.5.119,ClientNodeID=135.13.3.114”
       “LogicalPortIDs=1230&2&33210”
     “NSAPTNAADDR=13F323EFCA45DABC34698723FEAC45CF289364ED,
ClientNodeID=135.13.3.118”
       “LogicalPortIDs=231&230987&3"
    "LSAAdvertisedRouter=135.13.3.113,TNACount=2,EDGE"
     “IPV4TNAADDR=135.19.24.170”
       “LogicalPortIDs=27”
     “NSAPTNAADDR=14A323EFCA45CABC34698823FEAC45CF289364FD”
       “LogicalPortIDs=31&38198&35"
;
```

Note: The RTRV NADR ALL command can produce large amounts of data. This will require proper sizing of the receive buffer of the OS system.

Example output for the **RTRV-NADR** command with LSAID

The **RTRV-NADR** command with *dataType* = LSAID produces this example output result:

```
RTRV-NADR:LT-ONNS-165::123456::LSAID,135.13.3.113;

   LT-ONNS-165 2001-01-14 09:45:28
M  123456 COMPLD
   ″NADR,LSAID=135.13.3.113,135.13.3.113,NumberOfLSAs=2″
   ″LSAAdvertisedRouter=135.13.3.113,TNACount=3,UNI″
     ″IPV4TNAADDR=135.16.3.120,ClientNodeID=135.13.3.114″
       ″LogicalPortIDs=320″
     ″IPV4TNAADDR=176.113.5.119,ClientNodeID=135.13.3.114″
       ″LogicalPortIDs=1230&2&33210″
     ″NSAPTNAADDR=13F323EFCA45DABC34698723FEAC45CF289364ED,
      ClientNodeID=135.13.3.118″
       ″LogicalPortIDs=231&230987&3″
     ″LSAAdvertisedRouter=135.13.3.113,TNACount=2,EDGE″
     ″IPV4TNAADDR=135.19.24.170″
       ″LogicalPortIDs=27″
     ″NSAPTNAADDR=14A323EFCA45CABC34698823FEAC45CF289364FD″
       ″LogicalPortIDs=31&38198&35″
;
```

Error responses

Validation rule: Invalid componentID parameter I

The command is rejected when for the *datatype* = LSAID the received *ASTNNodeID* is invalid.

Error code: `E511 – IINO – Input, Invalid NOde_ID`

Validation rule: Access to the database is not possible I

Access to the database is not possible due to internal problems.

Error code:
`E002 – SICE – Status, Inter-Processor Communication Error caused cmd failure`

Validation rule: Access to the database is not possible II

Access to the database is not possible due to internal problems.

Error code: `E003 – SNTT – Status, Network Topology database Timed out`

Related TL1 messages

- "RTRV-L1ADR: Retrieve level 1 Address Data Repository" (p. 3-2439)

RTRV-NE: Retrieve NE

The **RTRV-NE** command retrieves DCN attributes associated with the NE at the system level but not associated with any particular facility or equipment unit.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-NE**:*tid*::*ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Description: Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Description: Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: CTAG.

Output format

If the **RTRV-NE** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   ":spec_block"
   .    .    .
   .    .    .
   .    .    .
   ":spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Spec block parameters

1. **ftamftpgw** – File Transfer Translation Device (FT-TD) function
   This parameter is used to enable or disable the File Transfer Translation Device (FT-TD) function.
   Type: Value set.
   Range: ENABLE, DISABLE (default).
2. **tl1tcpipgw** – TL1 Translation Device (T-TD) function
   This parameter is used to enable or disable the TL1 Translation Device (T-TD) function.
   Type: Value set.
   Range: ENABLE, DISABLE (default).
3. **dcnpn** – DCN ASAP Profile Name
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.
4. **scnpn** – SCN ASAP Profile Name
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: up to 24 characters.
   Default: DEFAULT.

Detailed behavior description

The command retrieves system parameters as described in the Output parameter section.

Example input/output

The following example shows the result of a **RTRV-NE** command:

```
RTRV-NE:LT-UNITE::123456;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
   ”:ftamftpgw=DISABLE”
   ”:tl1tcpipgw=DISABLE”
   ”:dcnpn=DEFAULT”
   ”:scnpn=DEFAULT”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-NE** command.

Related TL1 messages

"ED-NE: Edit NE" (p. 3-442)

RTRV-NE-SECU: Retrieve NE security

The **RTRV-NE-SECU** command can be initiated by a user to retrieve global security information in the NE. The current login sessions in the NE identified by the *uid* which established the session will also be retrieved using the **RTRV-NE-SECU** command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: No

Input format

Command syntax: **RTRV-NE-SECU***:tid::ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

There is one output line for each session. The first line contains user information for the user and session that issued the **RTRV-NE-SECU** command.

After receiving the **RTRV-NE-SECU** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″uid::spec_block″
   ″uid″
   .
   .
   ″uid″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **uid** – User ID
   ID of the user that initiated the **RTRV-NE-SECU** command. Listed below this *uid* are *uids* of all the other sessions. If the user from which the **RTRV-NE-SECU** command was issued is logged on via other sessions, the *uid* list will contain each other instance of the *uid*, but the additional listing is just like the listing of any other user, without the additional user information.

Spec block parameters

1. **age** – Lockout Aging Period
   The time interval during which the number of invalid login attempts are counted. When *age* expires after the last invalid login attempt, the counter of invalid login attempts by that *uid* is reset to zero.
   Type: Numeric string.
   Range: 1 – 999 [minute]
   Default: 60
2. **intrvl** – Lockout Period
   The time during which login attempts using that *uid* will not be allowed. This occurs when the number of consecutive invalid login attempts by a *uid* reaches *thrshld*. When *intrvl* expires, login attempts will again be allowed from that *uid*. If this parameter is set to 0 (zero), the lockout feature is disabled.
   Type: Numeric string.
   Range: 0 – 99 [minute]
   Default: 10
3. **thrshld** – Lockout Threshold
   The maximum number of consecutive invalid login attempts that are allowed by a *uid*. If the count of invalid login attempts is equal to the threshold, the *uid* is locked out for *intrvl* period of time. If a login attempt is successful, the count of *invalid* login attempts by that *uid* is reset to zero.
   Type: Numeric string.
   Range: 2 – 99
   Default: 5
4. **usrage** – User ID Aging Period
   If a *uid* has never been used to perform a successful login during a specified time interval it is disabled. A value of 0 disables user ID aging.
   The *usrage* does not apply to the super-users. Super-users will be never disabled.
   Type: Numeric string.
   Range: 0 – 999 [days]
   Default: 10

Example input/output

The following example shows the successful completion of the **RTRV-NE-SECU** command by the NE:

```
            RTRV-NE-SECU:LT-UNITE-123::123456;
   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
   ”nbguser1::thrshld=3,intrvl=1,age=60,usrage=10”
   ”njsmith”
   ”hugo”
   ”airmail”
;
```

Related TL1 messages

- “ACT-USER: Activate user” (p. 3-24)
- “ED-NE-SECU: Edit NE security” (p. 3-446)

RTRV-NNCNFG: Retrieve ONNS configuration data

Overview

The **RTRV-NNCNFG** command retrieves ONNS system configuration parameters (fixed and user-provisionable). It also retrieves ONNS software version information. The data is retrieved from the ONNS system configuration data repository.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command syntax: **RTRV-NNCNFG:***srcTid:[srcAid]:ctag;*

Input parameters

Command parameters

1. **srcTid** – Source Target Identifier.
   The TID of the NE to which the command is addressed.
   See the parameter description in the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) Input parameters section. NN uses the same *tid* used by the Host NE.
2. **srcAid**
   ONNS application access identifier: *snn*.
3. **ctag** – Correlation Tag
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
   It should be a unique value for each new command.

Output format

If the RTRV-NNCNFG request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   "specBlock"
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the OUTPUT PARAMETERS section of the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) command.

Spec block parameters

This block contains the descriptions and allowed values for the reported output parameters.

1. **`AdminCostEscalation`**
   Determines whether a cost escalation algorithm is applied or not. This algorithm raises the cost of a link bundle to reduce the attractiveness when the link bundle fills up.
   Type: Alphanumeric value set.
   Range: DISABLE, ENABLE (default).
2. **`DefaultRstrPriority`**
   Determines the connection restoration priority default to be used if a value is not specified in the **`ENT-PATH`** command. Lower values denote higher priority.
   Note: Though the parameter *DefaultRstrPriority* can be set via ED-NNCNFG, it is for future use only, i.e. an actual value different from 0 will not be considered on connection setup. If no restoration priority is given via the **`ENT-PATH`** command, the priorty of the corresponding connection is always 0, no matter of the actual value of *DefaultRstrPriority*.
   Type: Integer
   Range: 0 – 7 (default = 0, highest priority).
3. **`LBA_Threshold`**
   The LinkBundle Availability threshold determines the percent change in bandwidth availability necessary to trigger an immediate LSA distribution.
   Type. Integer
   Range: 10 – 65 %
   Default: 50 %
4. **`LBA_UpdateTimer`**
   LinkBundleAvailability Update Timer determines the maximum interval for LSA distribution of bandwidth availability changes which have not exceeded threshold.
   Type. Integer
   Range: 1 – 25 minutes
   Default: 15 minutes
5. **`LB_UsageThreshold`**
   The LinkBundle usage threshold determines the percent link bundle usage that triggers cost escalation on the bundle.
   Type. Integer
   Range: 50 – 95 %
   Default: 70 %

6. **`MaxExplicitNodeList`**
   This parameter determines the maximum number of intermediate nodes that will be accepted in the ENT-PATH explicit node list.
   Type. Integer
   Range: 25 (default).
7. **`MaxExcludeNodeList`**
   Determines the maximum number of nodes that will be accepted in the ENT-PATH exclude *nodeList*.
   Type. Integer
   Range: 25 (default).
8. **`MaxSRLGList`**
   Determines the maximum number of SRLG values that will be accepted in the ENT-PATH SRLG exclude list.
   Type. Integer
   Range: 25 (default).
9. **`NN_BldInfo`**
   Version of path calculation engine. Consists of:
   - NEtype
   - NN_SWVersion
   - Operating system

   Type. alphanumeric string
   Range: *ORPx.y* for NEtype, x, y digits, *LU-NNx.y.z* for NN_SWVersion, x, y, z digits, *Chorus* for Operating System
10. **`NN_BldVersion`**
    ONNS Build version and date in format *xxx-mmddyy*, since there is no difference any longer between the ONNS and host build version, this field is set to a default value. See corresponding parameters in RTRV-PRMTR-SWTFR and ACT-USER.
    Type: two – separated numeric strings
    Range: 000-000000 (default).
11. **`NNFunctionUpgradeState`**
    After a software upgrade, new ONNS functionality must be explicitly enabled. This parameter indicates whether this has been done or not.
    Type: Alphanumeric value set.
    Range:
    - DISABLE: New ONNS functionality not yet enabled
    - ENABLE: New ONNS functionality already enabled.

    Default: ENABLE.
12. **`NN_HostName`**
    Host name of the Host NE.
    Type: Alphanumeric string
    Range: 1675 LambdaUnite MSS (default).

13. **`NN_SigIpAddress`**
    Signaling IP address of the SCN.
    Type: Four dot separated integers.
    Range: 0 – 255 for each integer.
14. **`NN_SWVersion`**
    ONNS software version number in the format *x.y.z*, where
    - *x* = release,
    - *y* = point release,
    - *z* = development integration release.

    Type: Three dot separated integers.
    Range: 0 … 99 for each integer.
15. **`NN_SystemMode`**
    NN system mode status describes the operational state of ONNS.
    Type: Alphanumeric value set
    Range:
    - INIT: Initializing
    - MAINT: in maintenance condition (MCOND)
    - RUNNING: in normal operation
16. **`NN_UiIpAddress`**
    UI IP address (currently not used).
    Type: Four dot separated integers.
    Range: 0 – 255 for each integer.
17. **`NodeID`**
    Identifier of the ASTN node to which the **`RTRV-NNDR`** command is sent.
    Type: Four dot separated numeric string
    Range: 0 – 255 for each numeric string.
18. **`PathRestoreRetryMax`**
    Determines how many retries ONNS makes during the path restoration
    Type: Integer
    Range: 0 – 5
    Default: 1
19. **`PathSetupRetryMax`** – Maximum Path Restoration Retries
    Determines how many retries ONNS makes after the initial path setup.
    Type: Integer
    Range: 0 – 5
    Default: 1

20. **`Perm1plus1HoldOffTimer`** – Permanent 1+1 Hold Off Timer
    Determines how long a worker or protection path of a permanent 1+1 connection must be steadily failed before the failed path is restored non-revertively by a new path. Applies only to a permanent 1+1 connections.
    Type: Integer
    Range: 15 – 600 seconds in 1 second steps.
    Default: 360 seconds
21. **`RestorationRetryTimer`**
    Path Restoration Queue timer, sets the time between checks to restore from the queue.
    Type: Integer
    Range: 90 – 700 seconds
    Default: 324 seconds
22. **`TCAThresholdHigh`**
    Upper threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status HIGH for this link bundle
    Type: Integer
    Range: 10 – 100 %
    Default: 70 %
23. **`TCAThresholdLow`**
    Lower threshold in percent for the supervision of the unavailable bandwidth in the link bundles. Exceeding this threshold for a link bundle results in a TCA status LOW or HIGH for this link bundle.
    Type: Integer
    Range: 10 – 100 %
    Default: 40 %
24. **`tid`** – Node Target Identifier
    Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE.
    See also the definition of *tid* in RTRV-HDR.
    Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
    Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
25. **`WaitToRevertTimer`**
    Determines how long an original path must be steadily OK before it is made active again through tear-down of the restoration path. Applies to auto-reroute revertive connections. This parameter shall apply to M:N protected connections as well.
    Type: Integer
    Range: 60 – 1800 seconds
    Default: 300 seconds

Example input/output

The following is an example shows a successful execution of the **RTRV-NNCNFG** command:

```
RTRV-NNCNFG:DIETMAR:snn:C1;


   DIETMAR 2003-06-06 10:21:57
M  C1 COMPLD
   "TID = DIETMAR,
   NodeId = 29.54.174.87,
   NN_SWVersion = 2.0.0,
   NN_BldVersion = 042-032603,
   NN_BldInfo = ORP2.0, LU-NN2.0.0, Sun/Chorus,
   NN_HostName = LambdaUnite,
   NN_SystemMode = RUNNING,
   NN_SigIpAddress = 172.16.1.250,
   NN_UiIpAddress = 0.0.0.0,
   AdminCostEscalation = DISABLE,
   PathSetupRetryMax = 1,
   PathRestoreRetryMax = 1,
   DefaultRstrPriority = 0,
   RestorationRetryTimer = 324 seconds,
   Perm1plus1HoldOffTimer = 15 seconds,
   WaitToRevertTimer = 300 seconds,
   LBA_UpdateTimer = 15 minutes,
   LBA_Threshold = 50 %,
   LB_UsageThreshold = 70 %,
   TCAThresholdLow = 42 %,
   TCAThresholdHigh = 69 %,
   MaxExplicitNodeList = 8,
   MaxExcludeNodeList = 15,
   MaxSRLGList = 10,
   NNFunctionUpgradeState = ENABLE"
;
```

RTRV-NNDR: Retrieve ONNS neighbor nodal data repository

Overview

The **RTRV-NNDR** command retrieves information stored in the NNDR data repository that is in RAM memory. The **RTRV-NNDR** command operates on the current, active-memory-resident, ONNS data. ONNS writes port/connPt/pathID data to NVM shortly after each cross-connect setup of a path transaction completes. The NVM data is used in conjunction with port update messages from the NE during initialization to recover the ONNS NNDR data repository.

The NNDR record structure

The NNDR record is composed of an NNDR_HeaderRecord and multiple PortRecords which are described below, see the Output format and Output parameter sections for more detail.

Retrieving the NNDR records by ALL or by *portID*

ONNS responds to a RTRV-NNDR request to retrieve its NNDR by sending a report of the contents of the data repository. The command **RTRV-NNDR** with *dataType* = ALL shall cause the ONNS node to send all information in the NNDR data repository. When the NNDR is retrieved, each TL1 response message contains multiple PortRecords. If *dataType* = ALL, there must be no *componentID*. When *dataType* = PORT and a *componentID* = portID are specified as input parameters in RTRV-NNDR, the NNDR data repository records relating to that *portID* is returned

Retrieving the NNDR records by *neighborASTNNodeID* or *UNIclientNodeID*

When *dataType* = NBRIP and *componentID* = neighborASTNNodeID is specified, the response returns all NNDR PortRecords related to the link bundles between the reporting node and its neighbor node with the *neighborASTNNodeID*.

When *dataType* = NBRID and *componentID* = UNIClientNodeID is specified, the response returns all Neighbor Nodal data repository (NNDR) PortRecords related to the port connections between the reporting node and its neighbor node with the *UNIClientNodeID*.

The command will fail if the *neighborASTNNodeID* or the *UNIClientNodeID* is not a neighbor of the reporting node.

Retrieving the NNDR records by *PortUsage*

When *dataType* = PORTUSAGE and a *componentID* = portClass is specified, the response returns all NNDR records relating to ports with the *portClass* specified in the *componentID* field.

The **RTRV-NNDR** command sends a maximum of 4 PortRecords per block of data returned if there are no associated cross connects on the ports. PortRecords with *TribConnRecords* will be send in data blocks that are less than 4096 characters. On all messages but the last, a RTRV completion code is used and the semicolon (;) is replaced with a (>) terminator. Then, on the last record block, the COMPLD completion code is used with the final semicolon (;) terminator.

Data blocks in PortRecords with TribConnRecords shall contain complete TribConnRecords only including leading and trailing double quotes (″).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command syntax: **`RTRV-NNDR:`***srcTid:[srcAid]:ctag::dataType[,componentID];*

Input parameters

Command parameters

1. **`srcTid`** – Target Identifier.
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
4. **`dataType`**
   This parameter specifies, which part of the NNDR is to be retrieved. Furtheron it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.

Values:

- ALL: retrieves the entire NNDR in RAM.
- PORT: retrieves NNDR port data, *componentID* must be a *portID*.
- NBRID: retrieves all NNDR port data for link bundles and port connections respectively between the local and neighbor node, *componentID* must be a *neighborASTNNodeID* or a *UNIClientNodeID*.
- PORTUSAGE: retrieves all NNDR port data of ports of a specified port class, *componentID* must be a *portClass*.

5. **`ComponentID`**

   This parameter depends on the previous *dataType* parameter and is one of: not represented (when *datatype* = ALL), *portID* (when *datatype* = PORT, see definition of output parameter *portID* in RTRV-PATH), *nodeID* (when *datatype* = NBRID), *portClass* (when *datatype* = PORTUSAGE).

   The detailed parameter format of the choices is described below:

   - **`portID`** – Port Identifier

     Port ID for which the NNDR data is retrieved. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #.

     Type: dash-separated list of 4 numeric strings

     Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16).
   - **`nodeID`** – Neighbor Node Identifier

     This parameter identifies a neighbor ASTN node or a neighbor UNI client.

     Type: Four dot separated numeric strings (0 – 255 for each numeric string).
   - **`portClass`**

     This parameter specifies which type of port to include in the report, it determines how ONNS uses the port: EDGE (ONNS client port, where the signal enters/leaves the ONNS domain); INNI (ONNS port within the ONNS domain, topology relevant); ENNI (ONNS port, where the signal enters/leaves the ONNS domain from/to a other domain), UNI (ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client).

Output format

RTRV-NNDR output format without TribConnRecords

If the **`RTRV-NNDR`** request completes successfully, then the following normal NNDR_Report without TribConnRecords will be returned, when no tributary connections exists:

```
   srcTid date time
M  ctag RTRV
   ″NNDR_HeaderRecord″
   ″PR:PortStatusRecord_1,PortAttributeRecord_1″
   ...
   ″PR:PortStatusRecord_4,PortAttributeRecord_4″
>
```

```
   srcTid  date  time
M  ctag COMPLD
   ″PR:PortStatusRecord_5,PortAttributeRecord_5″
   ...
   ″PR:PortStatusRecord_8,PortAttributeRecord_8″
;
```

The **RTRV-NNDR** command will send a maximum of 4 port records per block. On all messages but the last, a RTRV completion code is used and the semicolon (;) is replaced with a (>) terminator. Then, on the last record block, the COMPLD completion code is used with the final semicolon (;) terminator.

RTRV-NNDR output format with TribConnRecords

If the **RTRV-NNDR** request completes successfully, then the following normal NNDR_Report with TribConnRecords will be returned, when there are tributary connections:

```
   srcTid date time
M  ctag RTRV
   ″NNDR,ALL,srcNodeASTNNodeID,NumberOfRecords=portCount″
   ″PR:PortStatusRecord1,PortAttributeRecord1″
   [″TR:TribPathRecord_1[:TribPathRecord_1b]″]
   ...
   [″TR:TribPathRecord_n[:TribPathRecord_nb]″]
   .....
   .....
   ″PR:PortStatusRecord4,PortAttributeRecord4″
   [″TR:TribPathRecord_1[:TribPathRecord_1b]″]
   [″TR:TribPathRecord_2[:TribPathRecord_2b]″]
   [″TR:TribPathRecord_3[:TribPathRecord_3b]″]
   [″TR:TribPathRecord_4[:TribPathRecord_4b]″]
>

   srcTid  date  time
M  ctag COMPLD
   ″PR:PortStatusRecord5,PortAttributeRecord5″
   [″TR:TribPathRecord_1[:TribPathRecord_1b]″]
   ...
   [″TR:TribPathRecord_n[:TribPathRecord_nb]″]
   .....
   .....
   ″PortStatusRecord8,PortAttributeRecord8″
   [″TR:TribPathRecord_1[:TribPathRecord_1b]″]
   [″TR:TribPathRecord_2[:TribPathRecord_2b]″]
   [″TR:TribPathRecord_3[:TribPathRecord_3b]″]
   [″TR:TribPathRecord_4[:TribPathRecord_4b]″]
   [″TR:TribPathRecord_5[:TribPathRecord_5b]″]
```

```
;
```

Output syntax example with normal data retrieve report for one port with cross-connections

If the **RTRV-NNDR** request completes successfully, then the following port retrieve response is returned. The response is shown in terms of the variables that form the response record:

```
   srcTid  date  time
M  ctag COMPLD
   ″NNDR,dataType=componentID,srcNodeASTNNodeID,Number0fRecords=portCount″

      ″PR:portID,portClass,pdir,neighborNodeID&neighborPortID,lineSigStatus,portEqptStatus,pe

      portRate,monitorCap,propagationDelay,Cost=adminCost,,SRLG=srlgList,IncludeID=linkInclu
   ″TR:tribID1,pathID1,connPtA1,-X-,connPtB1,RsvStatus=rsvStatus1,XCStatus=
      xcStatus1,tribWidth1,tribSigStatus1
   :tribID1b,pathID1b,connPtA1b,-X-,connPtB1b,RsvStatus=rsvStatus1b,XCStatus=
      xcStatus1b,tribWidth1b,tribSigStatus1b″
   .........
   ″TR:tribIDn,pathIDn,xconnAn,-X-,xconnBn,RsvStatus=rsvStatusn,XCStatus=
      xcStatusn,tribWidthn,tribSigStatusn″
;
```

The *portCount* = n is shown in this example indicating that there are n tributary paths on this port.

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completionCode* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

NNDR_HeaderRecord

1. **nndrID**
   ONNS Data Repository *dataID* to be returned: NNDR, Neighbor Nodal data repository in RAM.
2. **dataType**
   This parameter specifies, which part of the NNDR is to be retrieved. Furtheron it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Values:
   - ALL: retrieves the entire NNDR in RAM.
   - PORT: retrieves NNDR port data, *componentID* must be a *portID*.

- NBRIP: retrieves all NNDR port data for link bundles and port connections respectively between the local and neighbor node, *componentID* must be a *neighborASTNNodeID* or a *UNIClientNodeID*.
- PORTUSAGE: retrieves all NNDR port data of ports of a specified port class, *componentID* must be a *portClass*.

3. **`componentID`**
   This parameter depends on the previous *dataType* parameter and is one of: not represented (when *datatype* = ALL), *portID* (when *datatype* = PORT, see definition of output parameter *portID* in RTRV-PATH), *nodeID* (when *datatype* = NBRIP), *portClass* (when *datatype* = PORTUSAGE).

   The detailed parameter format of the choices is described below:

   - **`portID`** – Port Identifier

     Port ID for which the NNDR data is retrieved. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #.

     Type: dash-separated list of 4 numeric strings

     Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16).
   - **`nodeID`** – Neighbor Node Identifier

     This parameter identifies a neighbor ASTN node.

     Type: Four dot separated numeric strings (0 – 255 for each numeric string).
   - **`portClass`**

     This parameter specifies which type of port to include in the report, it determines how ONNS uses the port: EDGE (ONNS client port, where the signal enters/leaves the ONNS domain), INNI (ONNS port within the ONNS domain, topology relevant), ENNI (ONNS port, where the signal enters/leaves the ONNS domain from/to a other domain), UNI (ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client).
4. **`srcNodeASTNNodeID`** – Source Node ASTN Node ID
   Identifier of the ASTN node to which the **`RTRV-NNDR`** command is sent.
   Range:Four dot separated numeric strings, 0 – 255 for each numeric string.
5. **`NumberOfRecords`**
   This parameter indicates the number of PortRecords returned in this query.
   Values: number of NNDR records 0 – 1000.

PortRecord - PortStatusRecord

Two PortStatusRecords are present per reported port, one for each direction (see parameter pdir below).

1. **`portID`** – port identifier
   portID for which the NNDR data is retrieved. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #.
   Type: dash-separated list of 4 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16).
2. **`portClass`**
   The *portClass* specifies how ONNS uses the port.

Values:

- INNI, ONNS port within the ONNS domain, topology relevant, bandwidth on this port is managed by ONNS
- EDGE, ONNS client port, where the signal enters/leaves the ONNS domain, unmanaged by ONNS, but used as the source or destination port of an ONNS connection.
- ENNI, ONNS port, where the signal enters/leaves the ONNS domain from/to another domain
- UNI, ONNS client port, where the signal enters/leaves the ONNS domain from/to a UNI client, used as the source or destination port of an ONNS connection
- NA, Traditional (non-NN) port, they are normally not included in the ONNS data repository (exception: connections with EDGE ports exist).

3. **`pdir`** – Port Direction
   This parameter specifies whether the PortRecord is for the port in receive or transmit direction.

   Values:

   - TX: PortRecord is for transmit side of port
   - RX: PortRecord is for receive side of port

4. **`neighbor`** – Neighbor Information

   The neighbor information consists of &-separated *neighborNodeID* and *neighborPortID* parameter.

   - **`neighborNodeID`**

     Identifier of the peer node (neighbor node) with respect to output parameter *portID*. This peer node can be a peer ASTN node or a peer UNI client node. The special value 0.0.0.0 indicates that the neighbor node ID is unknown. The *neighborNodeID* shall get the value “0.0.0.0” if the neighbor node is node of another domain.

     Type: Four dot separated numeric strings, (0 – 255) for each numeric string

   - **`neighborPortID`**

     PortID for which the NNDR data is retrieved. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #..

     Type: dash-separated list of 4 numeric strings.

     Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16).

     The special value 0-0-0-0 indicates that the neighbor *portID* of an ASTN node is unknown. The *neighborPortID* shall get the value “0-0-0-0” if the neighbor node is a UNI client node or a node of an other domain.

5. **`lineSigStatus`** – Port Line Signal Status
   Line Signal Status indicates the fault status of the line signal on the port as reported to ONNS from the Host..

Values:

- GOOD (ONNS assumes that the port can be used for transmission)
- BAD (ONNS assumes that the port can not be used for transmission)
- UNKNOWN (ONNS couldn’t derive status information from the Host).

6. **`portEqptState`**
This parameter indicates the fault status of the equipment associated with the port as reported to ONNS by the Host..
Values: OK (ONNS assumes that the equipment is ok and that the associated port can be used for transmission), FAILED (ONNS assumes that the equipment is failed and that the associated port can not be used for transmission), UNKNOWN (ONNS couldn’t derive status information from the Host).

7. **`portAdminMode`** – Port Control
This attribute determines the mode of operation of this port.
Values: NORMAL (Port is for unrestricted use by ONNS), LOCKOUT (ONNS is not allowed to use the port for new connections, note: ONNS might delete existing connections from that port).

PortRecord - PortAttributeRecord

Two PortAttributeRecords are present per reported port, one for each direction (see parameter *pdir* above).

1. **`portRate`**
This parameter specifies the bandwidth / port rate of the port.
Range:
   - STS3 (STM1E, STM1, OC3 port)
   - STS12 (STM4, OC12 port)
   - STS48 (STM16, OC48 port)
   - STS192 (STM64, OC192 port)
   - GE1 (GE1 port)
   - GE10 (GE10 port)
   - STS3-adaptive (STM1E, STM1, OC3 iNNI port; OC3 EDGE or eNNI port in adaptive mode)
   - STS12-adaptive (STM4, OC12 iNNI port; OC12 EDGE or eNNI port in adaptive mode)

- STS48-adaptive (STM16, OC48 iNNI port; OC48 EDGE or eNNI port in adaptive mode)
- STS192-adaptive (STM64, OC192 iNNI port; OC192 EDGE or eNNI port in adaptive mode)

2. **`propagationDelay`**
Propagation delay in milliseconds through the port connection between the pair of ports on the path. The time needed for a signal to traverse from this node to the neighbor node. A propagation delay of 1 milliseconds corresponds to about 450 km of fiber length.
Range: 0 – 255, default = 0.
Propagation delay parameter is not applicable for EDGE, ENNI and UNI ports and is set to the default value.
3. **`Cost`** – Administrative Cost
Provisioned administrative cost of the port connection when used in a connection path.
Range: 1 – 65535
Administrative cost parameter is not applicable for EDGE ports and is set to the default value. Administrative cost parameter is not applicable for UNI ports and is set to the default value.
4. **`SRLG`** – SRLG list
Shared Risk Link Group Identifier list. See also parameter srlgList in RTRV-rr. The max. number of identities that can be provisioned per port may be release dependent.
Type: Up to ten &-separated integers
Examples:
198&233&0&0&0&0&0&0&0&0
10&20&30&40&0&0&0&0&=&0&0
89&1000&4&10&20&31&111&231&300&777
Range: Each value: 0 – 65535.
Default: 0, means no SRLG provisioned.
SRLG list is not applicable for EDGE ports and is set to the default value. SRLG list is not applicable for UNI ports and is set to the default value.
5. **`IncludeID`** – VPN ID
Values: 0 – 65535. default: = 0, means *includeID* provisioned.
VPN ID is not applicable for EDGE, ENNI and UNI ports and is set to the default value.
6. **`XCRate`** (minXCrate)- Cross-connection rate
The XCRate consists of minXCrate&maxXCrate with
   - *minXCrate* denotes the smallest signal rate granularity used to provision a port into trib's,
   - *maxXCrate* denotes the largest signal rate granularity used to provision a port into trib's.

   Type: two &-separated alphanumeric value sets

Range *minXCrate*:

- STS1 (for all STM***m***, OC***n*** ports)
- STS3 (for all STM***m***, OC***n*** ports)
- STS12 (for all STM***m***, OC***n*** ports, with m = 4, 16, 64 and ***n*** = 12, 48, 192)
- STS48 (for all STM***m***, OC***n*** ports, with m = 16, 64 and ***n*** = 48, 192)
- STS192 (for all STM***m***, OC***n*** ports, with ***m*** = 64 and ***n*** = 192)
- STS1 (for 1 GbE ports in SONET mode)
- STS3 (for 1 GbE ports in SDH mode)
- STS1(for 1 GEPL ports in SONET mode)
- STS3 (for 1 GEPL ports in SDH mode)
- STS3 (for 10GEPL ports)

Range *maxXCrate*:

- STS1 (for all STM***m***, OC***n*** ports)
- STS3 (for all STM***m***, OC***n*** ports)
- STS12 (for all STM***m***, OC***n*** ports, with ***m*** = 4, 16, 64 and ***n*** = 12, 48, 192)
- STS48 (for all STM***m***, OC***n*** ports, with ***m*** = 16, 64 and ***n*** = 48, 192)
- STS192 for all STM***m***, OC***n*** ports, with ***m*** = 64 and ***n*** = 192)
- STS1 (for 1 GbE ports in SONET mode)
- STS3 (for 1 GbE ports in SDH mode)
- STS1(for 1 GEPL ports in SONET mode)
- STS3 (for 1 GEPL ports in SDH mode)
- STS3 (for 10GEPL ports)

7. **BundleID** – Link Bundle identifier

   This parameter is a unique numeric ID assigned to a link bundle or TNA address on this node.

   There are two preconditions for link bundling:

   - A number of parameters with same values, e.g. propagationDelay, provisioned on iNNI ports. See **RTRV-*rr***.
   - Same TNA address, provisioned as network TNA Address on EDGE ports or SDH/SONET UNI ports or as EPL TNA Address on GbE UNI ports. See **RTRV-*rr*** or “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).

   Range: 1 .. 32767 (valid BundleID), 0 (for EDGE ports without network TNA address and for eNNI ports).
   Type: Integer.

TribPathRecord

For ports with *portClass* = INNI one TribPathRecord is present per cross-connection associated with the port. For ports with *portClass* = EDGE, *portClass* = UNI or *portClass* = ENNI two TribPathRecords are present per bridged cross-connection and one TribPathRecord is present per non-bridged cross-connection associated with the port.

1. **`tribID`** – Tributary Identifier
   Identifier of the tributary whose *pathID* and cross-connect information is included below. The *tribID* also denotes the timeslot of the reported cross-connection in the reported port.

   Range: 1 - n

   - with n=3 for STM1E, STM1, OC3 ports
   - with n=12 for STM4, OC12 ports
   - with n=48 for STM16, OC48 ports
   - with n=192 for STM64, OC192 ports
   - with n=21 for GE1 ports
2. **`pathID`** – Path Identifier
   The *pathID* of the associated cross-connection. The special value 0.0.0.0-0 indicates, that there is no associated pathID, i.e. the port has a non-ONNS cross-connection.
   Type: character string of length up to 20.
   Range: nodeID-Index; nodeID: four dot separated numeric strings: each string 0..255, Index: Numeric string.
   Example: 135.13.196.203-432165
3. **`connPtA`** – Connection Point A
   The *connPoint* on the reporting port of the cross-connection connPoint pair. The connPtA consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # .
   Type: dash-separated list of 5 numeric strings.
   Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
   Special values for slot/pack, port, and tributary
   Slot/Pack # #
   Port # #
   Tributary # #
   Value “#” for slot/pack, port, and tributary indicates default tributary AID used for reporting the open end in not existing cross-connections in the A-node or Z-node for a protecting NN connection of an M:N protected connection group.
   Slot/Pack # *
   Port # *
   Tributary # *
   Value “*” for slot/pack, port, and tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified

in parameter *connPtA*) is allocated by one or more LO traditional cross-connections.
Slot/Pack # @
Port # @
Tributary # @
Value “@” for slot/pack, port, and tributary indicates that the corresponding HO input and/or output tributary specified by *connPtA* is involved as EDGE-prt tributary in an OUC.
Slot/Pack # $
Port # $
Tributary # $
Value “$” for slot/pack, port, and tributary indicates a temporary reservation of an output tributary of an ENNI port needed for a network connection setup.

4. **`connInd`** – Connection Indicator
Seperator for better readability.
Range: -X- (cross-connection setup)

5. **`connPtB`**
The *connPoint* opposite to the reporting port of the cross-connection *connPoint* pair. The *connPtB* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #.
Type: dash-separated list of 5 numeric strings.
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128)
Special values for slot/pack, port, and tributary
Slot/Pack # #
Port # #
Tributary # #
Value “#” for slot/pack, port, and tributary indicates default tributary AID used for reporting the open end in not existing cross-connections in the A-node or Z-node for a protecting NN connection of an M:N protected connection group.
Slot/Pack # *
Port # *
Tributary # *
Value “*” for slot/pack, port, and tributary indicates unknown HO tributary AIDs used for reporting that the corresponding HO tributary of an EDGE port (specified in parameter *connPtA*) is allocated by one or more LO traditional cross-connections.
Slot/Pack # @
Port # @
Tributary # @
Value “@” for slot/pack, port, and tributary indicates that the corresponding HO input and/or output tributary specified by *connPtA* is involved as EDGE-prt tributary in an OUC.

Slot/Pack # $
Port # $
Tributary # $
Value "$" for slot/pack, port, and tributary indicates a temporary reservation of an output tributary of an ENNI port needed for a network connection setup.

6. **`RsvStatus`** – Reservation Status
   Indicator whether the cross-connection has been allocated to a path by ONNS..
   When *RsvStatus* = 0 and *XCStatus* = 1, this is a Host NE initiated cross-connection.
   Reserved (by ONNS) = 1; Unreserved = 0; default = 1.
7. **`XCStatus`**
   Indicator whether the cross-connect operation was successful.
   When *RsvStatus* = 0 and *XCStatus* = 1, this is a Host NE initiated cross-connection.
   Range: Success = 1 (default); Failed = 0
   2 = Reserved, for cross-connections, that are not physically switched, i.e. cross-connections in A- and Z-node for protecting NN connections within M:N protected connection groups.
   3 = Reserved, for cross-connections, that are not switched on higher order, but the corresponding HO tributary is blocked by a LO cross-connection on a substructure.
   4 = Reserved, for cross-connections, that are switched to/from an EDGE tributary involved as EDGE-prt in an OUC.
8. **`tribWidth`** – Tributary Width
   The used bandwidth of the cross-connections expressed in multiples of STS1s.
   Range: STS1, STS3, STS12, STS48, STS192.
9. **`tribSigStatus`** – Tributary Signal Status
   This parameter indicates the fault status of the *connPtA* incoming signal on a path.
   Range: OK, FAILED.

Note: In the reply to the TL1 command **`RTRV-NNDR`** the parameter *tribSigStatus* is missing. This parameter is sent per tributary path record being sent for all connected tributaries of all ports reported and gives the state of the incoming signal at the corresponding tributary (value OK or FAILED). As tributary record information from RTRV-NNDR is not displayed at the GUI, this issue is only recognized when checking TL1 logfiles. Workaround: To get state information at least for paths as a whole, RTRV-PATH can be executed or the corresponding GUI screens can be used.

Example input/output

Example NNDR output for the `RTRV-NNDR` command with an INNI *portID*:

The following example shows the output of a **`RTRV-NNDR`** command for an INNI portID:

```
RTRV-NNDR:LT-ONNS-319::123456::PORT,1-1-3-4;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   "NNDR,PORT=1-1-3-4,172.41.18.0,NumberOfRecords=2"
```

```
   "PR:1-1-3-4,INNI,TX,172.41.19.1&1-1-1-2,,OK,OK,,NORMAL,,

      STS48,MON,10,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=39"
   "TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:13,172.3.19.0-1251,1-1-3-4-13,-X-,
       1-1-6-1-49,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:25,172.3.19.65-51,1-1-3-4-25,-X-,1-1-3-1-7,RsvStatus=1,XCStatus=1,STS3,OK"
   "TR:28,172.3.19.53-91,1-1-3-4-28,-X-,1-1-3-4-4,RsvStatus=1,XCStatus=1,STS3,OK"

   "PR:1-1-3-4,INNI,RX,172.41.19.1&1-1-1-2,,OK,OK,,NORMAL,,

      STS48,MON,10,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=39"
   "TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK"
   "TR:13,172.3.19.0-1251,1-1-3-4-13,-X-,
       1-1-6-1-49,RsvStatus=1,XCStatus=1,STS48,OK"
   "TR:25,172.3.19.65-51,1-1-3-4-25,-X-,1-1-3-1-7,RsvStatus=1,XCStatus=1,STS3,OK"
   "TR:28,172.3.19.53-91,1-1-3-4-28,-X-,1-1-3-4-4,RsvStatus=1,XCStatus=1,STS3,OK"
;
```

Newlines between PortRecords are optional. The label -X- indicates the cross-connection between connPtA and connPtB on the TribPathRecord.

Example NNDR output for the `RTRV-NNDR` command with an INNI portID with protecting connection involved

The following example shows the output of a **`RTRV-NNDR`** command for an INNI portID with one tributary being allocated for a protecting connection in the A-node for this connection:

```
RTRV-NNDR:LT-ONNS-319::123456::PORT,1-1-3-4;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   "NNDR,PORT=1-1-3-4,172.41.18.0,NumberOfRecords=2"
   "PR:1-1-3-4,INNI,TX,172.41.19.1&1-1-1-2,,OK,OK,,NORMAL,,

      STS48,MON,10,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=39"
   "TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:13,172.3.19.0-1251,1-1-3-4-13,-X-,
       1-1-6-1-49,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:25,172.3.19.65-51,1-1-3-4-25,-X-,1-1-#-#-#,RsvStatus=1,XCStatus=2,STS3,OK"
   "TR:28,172.3.19.53-91,1-1-3-4-28,-X-,1-1-3-4-4,RsvStatus=1,XCStatus=1,STS3,OK"

   "PR:1-1-3-4,INNI,RX,172.41.19.1&1-1-1-2,,OK,OK,,NORMAL,,

      STS48,MON,10,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=39"
   "TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:13,172.3.19.0-1251,1-1-3-4-13,-X-,
       1-1-6-1-49,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:25,172.3.19.65-51,1-1-3-4-25,-X-,1-1-#-#-#,RsvStatus=1,XCStatus=2,STS3,OK"
```

```
   ”TR:28,172.3.19.53-91,1-1-3-4-28,-X-,1-1-3-4-4,RsvStatus=1,XCStatus=1,STS3,OK”
;
```

Example NNDR output for the `RTRV-NNDR` command with an EDGE *portID*

The following example shows the output of a **`RTRV-NNDR`** command for an EDGE *portID*:

```
RTRV-NNDR:LT-ONNS-319::123456::PORT,1-1-3-4;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   ”NNDR,PORT=1-1-3-4,172.3.19.0,NumberOfRecords=2”
   ”PR:1-1-3-4,EDGE,TX,0.0.0.0&0-0-0-0,,OK,OK,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0”
   ”TR:1,172.3.19.0-211,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK”
   ”TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-3-2-1,RsvStatus=1,XCStatus=1,STS48,OK”

   ”PR:1-1-3-4,EDGE,RX,0.0.0.0&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0”
   ”TR:1,172.3.19.0-211,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK”
   ”TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-3-2-1,RsvStatus=1,XCStatus=1,STS48,OK”
;
```

This example shows two TribPathRecords for two bridged paths of a connection on an EDGE port.

Example NNDR output for the RTRV-NNDR command with an EDGE portID of EDGE port configured for adaptive mode

The following example shows the output of a **`RTRV-NNDR`** command for an EDGE *portID* in adaptive mode:

```
RTRV-NNDR:LT-ONNS-319::123456::PORT,1-1-3-4;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   ”NNDR,PORT=1-1-3-4,172.3.19.0,NumberOfRecords=2”
   ”PR:1-1-3-4,EDGE,TX,0.0.0.0&0-0-0-0,,OK,OK,,NORMAL,,
   STS48-
      adaptive,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID
   ”TR:1,172.3.19.0-211,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK”
   ”TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-3-2-1,RsvStatus=1,XCStatus=1,STS48,OK”

   ”PR:1-1-3-4,EDGE,RX,0.0.0.0&0-0-0-0,,OK,UNKNOWN,,NORMAL,,
   STS48-
      adaptive,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID
   ”TR:1,172.3.19.0-211,1-1-3-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK”
```

```
   ″TR:1,172.3.19.0-1011,1-1-3-4-1,-X-,1-1-3-2-1,RsvStatus=1,XCStatus=1,STS48,OK″
;
```

Example NNDR output for the RTRV-NNDR command with a UNI portID

The following example shows the output of a **RTRV-NNDR** command for an UNI portID:

```
RTRV-NNDR:LT-ONNS-319::123456::PORT,1-1-4-4;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   ″NNDR,PORT=1-1-4-4,172.3.19.0,NumberOfRecords=1″
   ″PR:1-1-4-4,UNI,TX,192.41.19.1&0-0-0-0,,OK,OK,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3″
   ″TR:1,172.3.19.0-211,1-1-4-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK″

   ″PR:1-1-4-4,UNI,RX,192.41.19.1&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3″
   ″TR:1,172.3.19.0-211,1-1-4-4-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
  ;
```

Example NNDR output for the RTRV-NNDR command with a neighbor ASTN node ID

The following example shows the output of a **RTRV-NNDR** command with an input neighborASTNNodeID:

```
RTRV-NNDR:LT-ONNS-319::123456::NBRID,172.3.19.69;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   ″NNDR,NBRID=172.3.19.69,172.3.19.109,NumberOfRecords=8″
   ″PR:1-1-3-1,INNI,TX,172.3.19.69&1-2-5-1,,OK,UNKNOWN,,NORMAL,,

      STS192,MON,10,Cost=1,,SRLG=5&7&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleID=11
   ″TR:1,182.63.219.10-219,1-1-3-1-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,192.168.7.12-9,1-1-3-1-49,-X-,1-1-5-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-3-1,INNI,RX,172.3.19.69&1-2-5-1,,OK,OK,,NORMAL,,

      STS192,MON,10,Cost=1,,SRLG=5&7&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleID=11
   ″TR:1,182.63.219.10-219,1-1-3-1-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,192.168.7.12-9,1-1-3-1-49,-X-,1-1-5-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   .......

   ″PR:1-1-6-1,INNI,TX,172.3.19.69&1-1-1-4,,OK,FAILED,,NORMAL,,

      STS192,TRANS,10,Cost=1,,SRLG=5&7&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleID=
   ″TR:1,182.63.219.10-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
```

```
   ″TR:49,192.168.3.5-54,1-1-6-1-49,-X-,1-1-5-1-49,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-6-1,INNI,RX,172.3.19.69&1-1-1-4,,OK,UNKNOWN,,NORMAL,,

      STS192,TRANS,10,Cost=1,,SRLG=5&7&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,Bundle
   ″TR:1,182.63.219.10-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,192.168.3.5-54,1-1-6-1-49,-X-,1-1-5-1-49,RsvStatus=1,XCStatus=1,STS48,OK″
;
```

The command **RTRV-NNDR** will send a maximum of 4 bidirectional port records per block. This produces a list of port records for all ports connected to the node with neighborASTNnodeID = 172.3.19.69 that may require blocking the data with the continuation RTRV completion code and replacing the semicolon (;) with a (>) terminator until the last block.

Example NNDR output for the RTRV-NNDR command with a neighbor ASTN node ID

The following example shows the output of a **RTRV-NNDR** command with an input UNIclientNodeID:

```
RTRV-NNDR:LT-ONNS-319::123456::NBRID,172.3.19.69;

   LT-ONNS-319 2002-01-14 09:45:28
M  123456 COMPLD
   ″NNDR,NBRID=172.3.19.69,172.3.19.109,NumberOfRecords=8″
   ″PR:1-1-3-1,UNI,TX,172.3.19.69&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS192,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleID=
   ″TR:1,172.3.19.109-219,1-1-3-1-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,172.3.19.109-9,1-1-3-1-49,-X-,1-1-5-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-3-1,UNI,RX,172.3.19.69&0-0-0-0,,OK,OK,,NORMAL,,

      STS192,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleID=
   ″TR:1, 172.3.19.109-219,1-1-3-1-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49, 172.3.19.109-9,1-1-3-1-49,-X-,1-1-5-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   .......

   ″PR:1-1-6-1,UNI,TX,172.3.19.69&0-0-0-0,,OK,FAILED,,NORMAL,,

      STS192,TRANS,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleI
   ″TR:1,172.3.19.109-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,172.3.19.109-54,1-1-6-1-49,-X-,1-1-5-1-49,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-6-1,UNI,RX,172.3.19.69&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS192,TRANS,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS192,BundleI
   ″TR:1,172.3.19.109-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″TR:49,172.3.19.109-54,1-1-6-1-49,-X-,1-1-5-1-49,RsvStatus=1,XCStatus=1,STS48,OK″
;
```

The command **RTRV-NNDR** will send a maximum of 4 bidirectional port records per block. This produces a list of port records for all ports connected to the node with UNIclientnodeID = 172.3.19.69 that may require blocking the data with the continuation RTRV completion code and replacing the semicolon (;) with a (>) terminator until the last block.

Example NNDR output for the RTRV-NNDR command with ALL

The next example shows the output for the **RTRV-NNDR** command with ALL. The **RTRV-NNDR** command will send a maximum of 4 bidirectional port records per block:

```
RTRV-NNDR:LT-ONNS-139::12345::ALL;

   LT-ONNS-39 2002-01-14 09:45:28
M  12345 COMPLD
   ″NNDR,ALL,172.3.19.0,NumberOfRecords=16″
   ″PR:1-1-1-1,INNI,TX,192.41.19.1&1-1-1-2,,OK,OK,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″TR:1,172.3.19.0-1011,1-1-1-1-1,-X-,1-1-2-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-1-1,INNI,RX,192.41.19.1&1-1-1-2,,OK,UNKNOWN,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″PR:1-1-1-2,INNI,TX,172.3.19.1&1-1-1-1,,FAILED,UNKNOWN,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″PR:1-1-1-2,INNI,RX,172.3.19.1&1-1-1-1,,OK,OK,,NORMAL,,

      STS48,TRANS,100,cost=4,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″TR:1,192.3.19.0-1199,1-1-1-2-1,-X-,1-1-2-2-1,RsvStatus=1,XCStatus=1,STS48,OK″

   ″PR:1-1-1-3,INNI,TX,172.3.19.1&1-1-3-4,,OK,OK,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″TR:1,172.3.19.0-1065,1-1-1-3-1,-X-,1-1-4-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-1-3,INNI,RX,172.3.19.1&1-1-3-4,,OK,FAILED,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″TR:1,172.3.19.0-1065,1-1-1-3-1,-X-,1-1-4-1-1,RsvStatus=1,XCStatus=1,STS48,OK″
   ″PR:1-1-1-4,INNI,TX,172.3.19.1&1-1-1-3,,OK,UNKNOWN,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3
   ″PR:1-1-1-4,INNI,RX,0.0.0.0&1-1-1-1,,OK,UNKNOWN,,NORMAL,,

      STS48,TRANS,100,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=3

   ″PR:1-1-2-1,EDGE,TX,0.0.0.0&1-1-1-1,,OK,UNKNOWN,,NORMAL,,

      STS48,MON,0,Cost=0,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0″
   ″PR:1-1-2-1,EDGE,RX,0.0.0.0&1-1-1-1,,OK,OK,,NORMAL,,
```

```
      STS48,MON,0,Cost=0,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0"
   "TR:1,172.3.19.0-1011,1-1-2-1-1,-X-,1-1-1-1-1,RsvStatus=1,XCStatus=1,STS48,OK"
   "PR:1-1-3-2,UNI,TX,0.0.0.0&1-1-1-1,,OK,OK,,NORMAL,,

      STS48,MON,0,Cost=0,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=93
   "TR:1,192.3.19.0-1199,1-1-1-2-2,-X-,1-1-1-2-1,RsvStatus=0,XCStatus=1,STS48,OK"
   "PR:1-1-3-2,UNI,RX,0.0.0.0&1-1-1-1,,OK,UNKNOWN,,NORMAL,,
   STS48,MON,100,Cost=0,,SRLG=0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=93"

   "PR:1-1-4-1,ENNI,TX,0.0.0.0&1-1-1-1,,OK,FAILED,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=57&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0
   "TR:1,172.3.19.0-1065,1-1-4-1-1,-X-,1-1-1-3-1,RsvStatus=1,XCStatus=1,STS48,OK"
   "PR:1-1-4-1,ENNI,RX,0.0.0.0&1-1-1-1,,OK,OK,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=57&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0
   "TR:1,172.3.19.0-1065,1-1-4-1-1,-X-,1-1-1-3-1,RsvStatus=1,XCStatus=1,STS48,OK"
   "PR:1-1-4-2,EDGE,TX,0.0.0.0&1-1-1-1,,FAILED,OK,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0"
   "PR:1-1-4-2,EDGE,RX,0.0.0.0&1-1-1-1,,FAILED,OK,,NORMAL,,

      STS48,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0"
;
```

Example NNDR output for the `RTRV-NNDR` command with a PORTUSAGE data type

The following example shows the output of a **`RTRV-NNDR`** command with an portClass input:

```
RTRV-NNDR:LT-ONNS-39::123456::PORTUSAGE,EDGE;

   LT-ONNS-39 2002-01-14 09:45:28
M  123456 COMPLD
   "NNDR,PORTUSAGE=EDGE,172.3.19.0,NumberOfRecords=4"
   "PR:1-1-3-1,EDGE,TX,0.0.0.0&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS192,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0
   "TR:1,182.63.219.10-219,1-1-3-3-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:49,182.63.219.10-599,1-1-3-1-49,-X-,1-1-5-2-
      1,RsvStatus=1,XCStatus=1,STS48,OK"

   "PR:1-1-3-1,EDGE,RX,0.0.0.0&0-0-0-0,,OK,NO-IN,,NORMAL,,

      STS192,MON,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0
   "TR:1,182.63.219.10-219,1-1-3-1-1,-X-,1-1-6-1-1,RsvStatus=1,XCStatus=1,STS12,OK"
   "TR:49,182.63.219.10-599,1-1-3-1-49,-X-,1-1-5-2-
      1,RsvStatus=1,XCStatus=1,STS48,OK"
```

```
   ”PR:1-1-6-1,EDGE,TX,0.0.0.0&0-0-0-0,,OK,OK,,NORMAL,,

      STS192,TRANS,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0”
   ”TR:1,182.63.219.10-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS12,OK”

   ”PR:1-1-6-1,EDGE,RX,0.0.0.0&0-0-0-0,,OK,UNKNOWN,,NORMAL,,

      STS192,TRANS,0,Cost=1,,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeID=0,XCRate=STS1&STS48,BundleID=0”
   ”TR:1,182.63.219.10-219,1-1-6-1-1,-X-,1-1-3-1-1,RsvStatus=1,XCStatus=1,STS12,OK”
;
```

Error responses

Validation rule: Invalid *PortID* or *componentID* parameter

A RTRV-NNDR request for a specific port ID with a port number < 1 or > 16 is denied with the error response

`Syntax, Parameter Data Invalid`

instead of

`E410 – Input, Invalid PortID`.

The following list identifies conditions that will cause a similar error response:

Error code: `E512 – Input, Invalid Neighbor node ID`

Related TL1 messages

- “ENT-PATH: Create ONNS path connection” (p. 3-1701)
- “ED-PATH: Add or modify alias of ONNS connection” (p. 3-739)
- “DLT-PATH: Delete ONNS path connection” (p. 3-222)
- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “REPT-NNEVT notification” (p. 3-2144)
- “ED-NNCNFG: Modify ONNS system configuration” (p. 3-450)
- “ENT-CONN: Establish network connections with TNA address parameters” (p. 3-1455)
- “DLT-CONN: Delete network connection” (p. 3-139)
- “REPT CONNCHG notification” (p. 3-2104)

Incorrect bandwidth reporting

In case a *unidirectional* Signal Degrade (SD) condition occurs on an INNI port connection within an ONNS domain, this SD condition leads to a near-end-failure condition being raised for the affected port connection by the detecting node for the corresponding direction only. This leads to the following behavior:

- For the direction opposite to the direction of the corresponding SD condition the bandwidth of the affected port connection is removed of the available bandwidth advertised for the corresponding INNI link bundle, although in this direction no failure condition at all is present.
- Asymmetric available bandwidth is reported for the affected INNI link bundle via **`RTRV-NTDR`**, but just the other way round as the actual failure condition would suggest.
- For the setup of a unidirectional ONNS path (due to connection setup or restoration) via the affected link bundle in the direction opposite to the SD condition the affected port connection will not be selected, although in this direction no failure condition exists and the setup would succeed.
- The setup of a unidirectional path in the direction of the SD condition or of a bidirectional path in any of both directions via the affected port connection would be possible and would succeed in first place, but might afterwards still fail with E022 due to the fact, that the SD condition on port connection level induced an SD condition on path level.
  This is due to the fact, that for path level an SD condition raises a failure condition that causes a path setup to fail.

Workaround

In case such asymmetric behavior occurs, INNI link bundles in the affected ONNS domain could be checked for SD conditions via **`RTRV-NTDR`**, **`RTRV-NNDR`**, and **`RTRV-rr`**. Additionally, an SD alarm should have been raised for the affected port, if not suppressed.

RTRV-NNEVT-TCA

Overview

The **RTRV-NNEVT-TCA** command retrieves the threshold control alert (TCA) status of the link bundles and operates on the current neighbor nodal data repository (NNDR) based local view on the bandwidth availability.

Retrieving all TCA status records

Network Navigator responds to a request to retrieve its TCA Status by sending a complete report of the corresponding contents of the NNDR for the local node.

The command **RTRV-NNEVT-TCA** ALL is supported, and it retrieves the entire TCA Status of the local node when the dataType field = ALL.

The **RTRV-NNEVT-TCA** command sends a maximum of 64 output lines per block. If possible, the next block shall start with the next LSA_Link_Record.

Retrieving TCA status by neighbor ASTN node ID

The command **RTRV-NNEVT-TCA** with dataType = NBRID and a neighborASTNNodeID as componentID retrieves all of the TCA status records that relate to the link between the local node and the neighbor node with that neighborASTNNodeID. The componentID is a single specific ASTN node ID of a neighbor node (ONNS adjacent node).

REPT PTHCHG or REPT NNCHG is not generated

The **RTRV-NNEVT-TCA** command do not generate a REPT PTHCHG or REPT NNCHG message.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

The **RTRV-NNEVT-TCA** command has a parameter dataType that defines what type of information is expected in the componentID field that determines what portion of the data repository records is retrieved.

Command syntax: **RTRV-NNEVT-TCA:***srcTid:[srcAid]:ctag::dataType[,componentID];*

Input parameters

Command parameters

1. **`srcTid`** – Source Target Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
4. **`dataType`**
   This parameter indicates the data input parameter type of the *componentID* in the next field. The input parameters in the next field may change with each *dataType* to suit the requirements of the requested *dataType*. ALL is supported in this field and requires no *componentID* in the next field.
   Type: Alphanumeric value set.
   Range:
   - *ALL*: retrieve all TCA Status information
   - *NBRID*: next field will contain a Neighbor ASTN node ID, not valid without a matching appropriate value in the *componentID* field except for dataType = ALL.
5. **`componentID`**
   This parameter allows the user to specify a portion of TCA Status information such as an ASTN node ID for a neighbor node to return the TCA Status for all bundles of a link. The TCA Status records for the specified *componentID* will be returned.
   componentID depends on the previous dataType parameter and is one of:
   - not represented when datatype=ALL, to get all TCA Status information
   - *nodeID*: when datatype=NBRID, to get the part of the TCA status of the link between that neighbor node and this node.
     Description of nodeID (Neighbor Node Identifier):

Identifier of a neighbor ASTN node or a neighbor UNI client.
Type: Four dot separated numeric strings.
Range: 0 … 255, for each numeric string.

Output format

The **RTRV-NNEVT-TCA** with *dataType* = ALL produces this example output result with the complete TCA status information of the local node:

```
RTRV-NNEVT-TCA:LT-ONN61:snn:123456::ALL;

   LT-ONN61 2002-01-14 09:45:28
M  123456 COMPLD
   ″nndrID,ALL,srcNodeASTNNodeID″
   ″LinkId=origNodeASTNNodeID&neighborASTNNodeID1″
    ″:BundleID=bundleID1,TCAStatus=TCA_Status1″
    ″:BundleID=bundleID3,TCAStatus=TCA_Status3″
   ″LinkId=origNodeASTNNodeID&neighborASTNNodeID9″
    ″:BundleID=bundleID9,TCAStatus=TCA_Status9″
   ″LinkId=origNodeASTNNodeID&neighborASTNNodeID2″
    ″:BundleID=bundleID2,TCAStatus=TCA_Status2″
    ″:BundleID=bundleID7,TCAStatus=TCA_Status7″
;
```

The **RTRV-NNEVT-TCA** command with *dataType* = NBRID produces this example output result:

```
RTRV-NNEVT-TCA:LT-ONN61:snn:123456::NBRID,neighborASTNNodeID;

   LT-ONN61 2002-01-14 09:45:28
M  123456 COMPLD
   ″nndrID,NBRID=neighborASTNNodeID,origNodeASTNNodeID″
   ″LinkId=origNodeASTNNodeID&neighborASTNNodeID″
    ″:BundleID=bundleID1,TCAStatus=TCA_Status1″
    ″:BundleID=bundleID4,TCAStatus=TCA_Status4″
    ″:BundleID=bundleID3,TCAStatus=TCA_Status3″
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completionCode* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

Header_Record

1. **`nndrID`** – ONNS Data Repository dataID to be returned
   Type: Alphanumeric value set.
   Range: NNDR-TCA: TCA portion of Neighbour Nodal data repository.
2. **`dataType`**
   See definition of *dataType* in Input parameters section above.
3. **`componentID`**
   See definition of *componentID* in Input parameters section above.
4. **`nodeID`** - Neighbor Node Identifier
   Identifier of a neighbor ASTN node or a neighbor UNI client.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0 … 255
5. **`origNodeASTNNodeID`** – ASTN node ID of the originating node.
   Type: Four dot separated numeric strings.
   Range: for each numeric string: 0..255.

Link_Header_Record

1. **`LinkID`**
   LinkID is the ASTN node ID of the originating node plus that of the neighbor node that uniquely identifies the link.
   Type: Two &-separated node IDs, each node ID consists of four dot separated numeric strings.
   Example: 135.13.44.61&135.13.3.86
   Range: for each numeric string: 0..255.

LinkBundle_Data Record

1. **`BundleID`**
   This parameter is a unique numeric ID assigned to the link bundle on this node..
   Type: Integer.
   Range: >0.
2. **`TCAStatus`** – Status of the TCA of this link bundle
   Type: Alphanumeric value set.
   Range: INACTIVE (default), LOW, HIGH

Example input/output

Output for the **`RTRV-NNEVT-TCA`** command by neighbor ASTN node ID.

The **`RTRV-NNEVT-TCA`** command with *datatype* = NBRID produces this example output result with all bundles between two neighbor nodes:

```
RTRV-NNEVT-TCA:LT-ONNS61:snn:123456::NBRID,135.13.44.86;

   LT-ONNS61 2001-01-14 09:45:28
M  123456 COMPLD
   ″NNDR-TCA,NBRID=135.13.44.86,135.13.44.61″
   ″LinkId=135.13.44.61&135.13.3.86″
```

```
    ":BundleId=1,TCAStatus=INACTIVE"
    ":BundleId=2,TCAStatus=LOW"
    ":BundleId=4,TCAStatus=HIGH"
;
```

The **RTRV-NNEVT-TCA** command with *datatype* = ALL produces this example output result:

```
RTRV-NNEVT-TCA:LT-ONNS-61:snn:123456::ALL;

   LT-ONNS-61 2002-07-14 09:45:28
M  123456 RTRV
   "NNDR-TCA,ALL,135.13.44.61"
   "LinkId=135.13.44.61&135.13.3.86"
    ":BundleId=1,TCAStatus=INACTIVE"
    ":BundleId=2,TCAStatus=LOW"
    ":BundleId=4,TCAStatus=HIGH"

....
   "LinkId=135.13.44.61&135.13.3.89"
    ":BundleId=3,TCAStatus=INACTIVE"
    ":BundleId=11,TCAStatus=LOW"
    ":BundleId=17,TCAStatus=LOW"
>
   LT-ONNS-61   2002-01-14   09:45:29
M  123456 RTRV
   "LinkId=135.13.44.61&135.13.3.95"
    ":BundleId=9,TCAStatus=INACTIVE"
    ":BundleId=13,TCAStatus=INACTIVE"
    ":BundleId=5,TCAStatus=INACTIVE"

....
   "LinkId=135.13.44.61&135.13.3.99"
    ":BundleId=19,TCAStatus=INACTIVE"
    ":BundleId=14,TCAStatus=LOW"
    ":BundleId=18,TCAStatus=LOW"

>    ....
   LT-ONNS-61   2002-01-14   09:45:30
M  123456 COMPLD
   "LinkId=135.13.44.61&135.13.3.105"
    ":BundleId=7,TCAStatus=LOW"
    ":BundleId=15,TCAStatus=LOW"
    ":BundleId=8,TCAStatus=HIGH"
;
```

The **RTRV-NNEVT-TCA** command will send a maximum of 64 lines per block.

Error responses

Validation rule: Invalid *componentID* parameter

The validity of the *componentID* parameter is checked. If it is invalid an appropriate error response is returned.

The following error responses is returned for an ASTN node ID which is not that of a neighbor:

Error code: `E512 – IINI Error – Input, Invalid Neighbor node ID`

Related TL1 messages

- "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-PTHEVT notification" (p. 3-2175)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT-NNEVT notification" (p. 3-2144)

RTRV-NTDR: Retrieve ONNS network topology data repository

Overview

The **RTRV-NTDR** command retrieves information stored in the NTDR data repository that is in RAM memory. The **RTRV-NTDR** command operates on the current, active-memory-resident, ONNS data. The NVM data from NNDM is used during initialization to recover the ONNS local NTDR data repository. Data from other ONNS nodes is acquired by the flooding and exchange of LSAs.

Retrieving all NTDR records

When the *dataType* = ALL, Network Navigator responds to a request to retrieve its NTDR by sending a complete report of the contents of the data repository specified. The request causes the ONNS node to send all information in the specified data repository. When the NTDR is retrieved, each TL1 response message contains a record with all of the data relevant to a ONNS link group or LSA. The **RTRV-NTDR** command will send a maximum of 14 LSA linkBundle records per block.

Retrieving NTDR records for any ASTN node ID

The command **RTRV-NTDR** with *dataType* = LSAID and *componentID* = ASTNNodeID retrieves the NTDR LSA records that relate to all of the links from the specified node with the *ASTNNodeID* to adjacent nodes. This *ASTNNodeID* can be the address of any node in the ONNS domain.

Retrieving NTDR records for a neighbor ASTN node ID

The command **RTRV-NTDR** with *dataType* = NBRID and *componentID* = neighborASTNNodeID retrieves all of the NTDR LSA records that relate to the links between the current node and the neighbor node with the *neighborASTNNodeID*. The *componentID* is a single specific ASTNNodeID. The *neighborASTNNodeID* must be the ID of a neighbor node (ONNS adjacent node).

REPT PTHCHG or REPT NNCHG is not generated

The **RTRV-NTDR** command shall not generate a REPT PTHCHG or REPT NNCHG notification.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No

Input format

Command syntax: **RTRV-NTDR**:*srcTid:srcAid:ctag::dataType[,componentID];*

Input parameters

Command parameters

1. **`srcTid`** – Source Target Identifier
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set.
2. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
4. **`dataType`**
   This parameter specifies, which part of the NNTR is to be retrieved. Furtheron it determines the expected input parameter type of the componentID input parameter. When datatype = ALL componentID must not be specified.

   Values:

   - ALL: retrieves the entire NTDR in RAM.
   - LSAID: retrieves the LSA in NTDR related to the ASTN node ID, *componentID* must be an *ASTNnodeID*
   - NBRID: retrieves the LSA in NTDR between the current and the specified neighbor node, *componentID* must be a *neighborASTNNodeID*.
5. **`componentID`**

   This parameter depends on the previous *dataType* parameter and is one of:

   - not represented (when *datatype* = ALL)
   - *nodeID* (when *datatype* = LSAID (any node ID) or *datatype* = NBRID (neighbornode ID)

   The details for *nodeId* are described below.

nodeID – Node Identifier

- Identifier of an ASTN node.
- Type: Four dot separated numeric strings
- Range: for each numeric string: 0..255.

Note: If *dataType* = ALL, *componentID* and the = sign are not reported.

Output format

The Neighbor Topology data repository (NTDR) record structure

The complete NTDR of one node consists of LSARecords for all known nodes (including the reporting node itself) in the topology. Within a LSARecord level 1 Routing Controller information is kept in L1Routing and all (direct) neighborship relationships are kept in LinkRecords (one record for each neighbor relation). Within a LinkRecord the individual LinkBundleRecords are listed.

Neighbor Topology data repository (NTDR) without data

An addressed node without data in the NTDR shall response only the HeaderRecord with NumberOfLSAs = 0. The Neighbor Topology data repository (NTDR) Record is composed of an NTDR_HeaderRecord and multiple LSARecords which are described below:

NTDR_Record ::=

<NTDR_HeaderRecord>

<LSARecord_1>

...

<LSARecord_n>

NTDR_HeaderRecord ::=

nndrID,dataType[=componentID],srcNodeASTNNodeID,NumberOfLSAs=LSA_Count

LSARecord ::=

<LSAHeaderRecord>

<L1Routing>

<LinkRecord_1>

...

<LinkRecord_n>

LSAHeaderRecord ::=

LSAAdvertisedRouter=origNodeASTNNodeID,LinkCount=linkCount,nodeTid, UIipAddr=nodeUI_IPaddr,SIGipAddr=nodeIPaddr

L1Routing::=

L1Routing=routeFunc,RoutingAreaID=routeAreaID,ParentRoutingAreaID =prouteAreaID,L1RoutingControllerID=routeControlID,L1RoutingControllerIPAddr= routeControlIPAddr,AbstractModel=abstractModel,AbstractNodeID=abstractNodeID

LinkRecord ::=

<Link_HeaderRecord>

<LinkBundleRecord_1>

…

<LinkBundleRecord_o>

Link_HeaderRecord ::=

LinkId=LsaLinkID,BundleCount=bundleCount

LinkBundleRecord ::=

BundleID=linkBundleID,Delay=delay,Cost=adminCost,

SRLG=srlgList1,IncludeId=0,AtMinRate=minXCrate,

TotalBandwidth=totalBandwidth,AvailBandwidth=availableBandwidth,

AtMaxRate=maxXCrate,AvailBandwidth=availableBW_MaxXCrate

Output format with normal data retrieve completion report for dataType=ALL

If the RTRV-NTDR request completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″nndrID,ALL,srcNodeASTNNodeID,NumberOfLSAs=LSA_Count″
   ″LSAHeaderRecord_1::
   Link_HeaderRecord_1:
   LinkBundleRecord_1-1
   ...
   LinkBundleRecord_1-n″
   ...
   ″LSAHeaderRecord_4:L1Routing_4:
   Link_HeaderRecord_4:
   LinkBundleRecord_4-1″
>
   srcTid date time
M  ctag COMPLD
   ″LSAHeaderRecord_5::
   Link_HeaderRecord_5:
   LinkBundleRecord_5-1
   ...
   LinkBundleRecord_5-n″
```

```
   ...
   ”LSAHeaderRecord_m::
   Link_HeaderRecord_m:
   LinkBundleRecord_m-1
   ...
   LinkBundleRecord_m-n”
;
```

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completionCode* and their syntax are included in the section “Output parameters” of the “RTRV-PATH: Retrieve ONNS connection data” (p. 3-2698) command.

NTDR_HeaderRecord

1. **nndrID**
   NN Data Repository *dataID* to be returned
   Type: Alphanumeric value set.
   Range: NTDR, Network Topology data repository in RAM

2. **dataType**
   This parameter specifies, which part of the ELDR is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set.

   Range:

   - ALL: retrieves the entire external link data repository in RAM
   - LSAID: retrieves the LSA in ELDR related to the ASTN node ID, *componentID* must be an *ASTNnodeID*
   - NBRID: retrieves the LSA in NTDR between the current and the specified neighbor node, *componentID* must be a *neighborASTNNodeID*.

   This parameter specifies, which part of the NNTR is to be retrieved. Furthermore it determines the expected input parameter type of the *componentID* input parameter. When *datatype* = ALL *componentID* must not be specified.
   Type: Alphanumeric value set

   Range:

   - ALL: retrieves the entire (NTDR) Neighbor Topology data repository in RAM
   - LSAID: retrieves the LSA in NTDR related to the ASTN node ID, *componentID* must be an *ASTNnodeID*
   - NBRID: retrieves the LSA in NTDR between the current and the specified neighbor node, *componentID* must be a *neighborASTNNodeID*.

3. **componentID**

This parameter depends on the previous *dataType* parameter and is one of:

- not represented (when *datatype* = ALL)
- *nodeID* (when *datatype* = LSAID (any node ID) or *datatype* = NBRID (neighbornode ID)

The details for *nodeId* are described below.

nodeID – Node Identifier

- Identifier of an ASTN node.
- Type: Four dot separated numeric strings
- Range: for each numeric string: 0..255.

Note: If *dataType* = ALL, *componentID* and the = sign are not reported.
Note: If *dataType* = ALL, *componentID* and the "=" sign are not reported.

4. **srcNodeASTNNodeID** – Reporting Node ID
   ASTN node ID of the ONNS node replying to this command.
   Type: Four dot separated numeric strings
   Range: for each numeric string: 0..255
5. **LSA_Count**
   The parameter indicates the number of LSA records in the report. Also equals number of NE nodes in the network for *dataType* = ALL.
   Type: Integer
   Range:
   - >0: number of LSAs with topology data records
   - 0: no topology data at this node or within the network available

LSARecord

For each reported node of the topology one LSARecord is reported.

1. LSAHeaderRecord
   a. **origNodeASTNNodeASTNNodeID** – Node ID of Advertising Node
      This is the ASTN node ID of the node advertising this LSA data record.
      Type: Four dot separated numeric strings.
      Range: 0 – 255 for each numeric string.
   b. **linkCount**
      Number of reported links of *LSAAdvertisedRouter* towards other nodes in the network.
      Type: Integer
      Range: >0
   c. **nodeTid**
      Target identifier of the source node is the logical name of the Host NE. ONNS uses the same unique *tid* used by Host NE.
      See also the definition of *tid* in RTRV-HDR.
      Type: identifier, in addition the special character underscore "_" and the password characters are allowed.

Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.

2. L1Routing

   The L1Routing object is optional and only visible for the *LSAAdvertisedRouter* with the value PRIMARY for the parameter *routeFunc*.

   a. **`routeFunc`** – level 1 Routing Function

      routeFunc defines if the *LSAAdvertisedRouter* has level 1 routing controller functionality.

      Type: Alphanumeric value set.

      Range: PRIMARY (ONNS node is the primary level 1 routing controller)

   b. **`routeAreaID`** – Routing Area ID

      routeAreaID defines the identity of the routing area.

      Type: four dot separated numeric string

      Range: 0 – 255 for each numeric string.

   c. **`prouteAreaID`** – Parent Routing Area ID

      prouteAreaID defines the identity of the parent routing area.

      Type: four dot separated numeric string

      Range: 0 – 255 for each numeric string.

   d. **`routeControlID`** – level 1 Routing Controller ID

      routeControlID defines the identity of the level 1 routing controller.

      Type: four dot separated numeric string

      Range: 0 – 255 for each numeric string.

   e. **`routeControlIPAddr`** – level 1 Routing Controller IP Address

      routeControlIPAddr defines the IP address for this routing controller node as level 1 routing controller address.

      Type: four dot separated numeric string

      Range:

      For first numeric string: 0 … 223

      For second to fourth numeric string: 0 … 255

      The values 0.0.0.0 and 1.1.1.1 are excluded.

   f. **`abstractModel`** – Abstraction Model

      abstractModel defines the abstraction model used by the level 1 routing controller.

      Type: Alphanumeric value set.

      Range: ABSTRACTNODE

   g. **`abstractNodeID`**

      abstractNodeID is defined in case the *abstractModel* is ABSTRACTNODE. This parameter specifies the abstract node ID, which is advertised in the level 1 area.

      Type: four dot separated numeric string

      Range: 0 – 255 for each numeric string.

LinkRecord

For each neighborship of the *LSAAdvertisedRouter* one LinkRecord is reported.

1. Link_HeaderRecord
    a. **`LsaLinkID`**

    LsaLinkID consists of the advertising ASTN nodeID and a neighbor ASTN nodeID, which together uniquely identify a link. The two IDs are separated by a &.

    Type: Two times four dot separated numeric strings, separated by &.

    Example: 135.13.44.61&135.13.3.86

    Range: 0 – 255 for each numeric string.

    b. **`bundleCount`**

    Number of LinkBundles within the LinkRecord.

    Type: Integer

    Range: >0

2. LinkBundleRecord

    For each link bundle within the LinkRecord one LinkBundleRecord is reported.

    a. **`linkBundleID`**

    This parameter is a unique numeric ID assigned to the link bundle on this node.

    Range: 0 .. 32767

    b. **`delay`**- Propagation Delay

    Propagation delay in milliseconds through the port connection between the pair of ports on the path. The time needed for a signal to traverse from this node to the neighbor node. A propagation delay of 1 milliseconds corresponds to about 450 km of fiber length.

    Type: Integer

    Range: 0 (default) – 255

    Propagation delay parameter is not applicable for EDGE, ENNI and UNI ports and is set to the default value.

    c. **`adminCost`** – Administrative Cost

    Provisioned administrative cost of the port connection when used in a connection path.

    Type: Integer

    Range; 1 (default) – 65535

    Administrative cost parameter is not applicable for EDGE ports and is set to the default value. Administrative cost parameter is not applicable for UNI ports and is set to the default value.

    d. **`slrgList`** – Shared Risk Link Group Identifier list.

    The max. number of identities that can be provisioned per port may be release dependent.

    Type: Up to ten &-separated integers

    Examples:

    198&233&0&0&0&0&0&0&0&0

10&20&30&40&0&0&0&0&=&0&0

89&1000&4&10&20&31&111&231&300&777

Range: Each value: 0 (default) – 65535. 0 means no SRLG provisioned

SRLG list is not applicable for EDGE ports and is set to the default value. SRLG list is not applicable for UNI ports and is set to the default value.

e. **`IncludeID`** – VPN ID

   Type: Integer

   Range: 0 (default) – 65535. 0 means no includeID provisioned

   VPN ID is not applicable for EDGE, ENNI and UNI ports and is set to the default value.

f. **`minXCrate`**

   minXCrate denotes the smallest signal rate granularity used to provision a port into trib’s.

   Values: STS1 (link bundle consists of STMm, OCn ports), STS3 (link bundle consists of GE1 ports in SDH mode), STS1 (link bundle consists of GE1 ports in SONET mode).

g. **`totalBandwidth`**

   This parameter is a numeric value indicating the total bandwidth in this linkBundle as multiples of *minXCrate*.

   Type: Integer

   Range: >2

h. **`availableBandwidth`** (for *minXCrate*)

   This parameter is a numeric value indicating the available (i.e. non-used) bandwidth in this *linkBundle* as multiples of *minXCrate*.

   Type: Integer

   Range: ≥0

i. **`maxXCrate`**

   This parameter denotes the largest signal rate granularity used to provision a port into trib’s.

   Type: Alphanumeric value set.

   Range: STS1, STS3, STS12, STS48, STS192 (link bundle consists of all STMm, OCn ports), STS1 (link bundle consists of GE1 ports in SONET mode), STS3 (link bundle consists of GE1 ports in SDH mode).

j. **`availableBW_MaxXCrate`**

   This parameter is a numeric value indicating the available (i.e. non-used) bandwidth in this *linkBundle* as multiples of *maxXCrate*.

   Type: Integer

   Range: ≥0

Example input/output

Example output for the **`RTRV-NTDR`** command by Neighbor ASTN node ID:

```
RTRV-NTDR:LT-ONNS-61::123456::NBRID,135.13.44.86;
```

```
   LT-ONNS-61 2001-01-14 09:45:28
M  123456 COMPLD
   ″NTDR,NBRID=135.13.44.86,135.13.44.61,NumberOfLSAs=2″
   ″LSAAdvertisedRouter=135.13.44.61,LinkCount=1,LT-ONNS-61,
UIipAddr=135.3.9.61,SIGipAddr=135.18.64.79::
   LinkId=135.13.44.61&135.13.3.86,BundleCount=3:
     BundleId=1,Delay=10,Cost=1,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=288,AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
     BundleId=2,Delay=10,Cost=10,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=192,AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
     BundleId=4,Delay=20,Cost=5,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=384,AvailBandwidth=336,
     AtMaxRate=STS192,AvailBandwidth=1″

   ″LSAAdvertisedRouter=135.13.44.86,LinkCount=1,LT-ONNS-45,
UIipAddr=135.3.9.45,SIGipAddr=135.18.64.83:
   L1Routing=PRIMARY,RoutingAreaID=111.234.215.6,ParentRoutingAreaID=
111.214.115.7,L1RoutingControllerID=218.3.45.56,L1RoutingControllerIPAddr=
104.34.49.1,AbstractModel=ABSTRACTNODE,AbstractNode=129.239.4.23:
   LinkId=135.13.44.86&135.13.3.61,BundleCount=3:
     BundleId=2,Delay=10,Cost=1,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=288,AvailBandwidth=0,
     AtMaxRate=STS48,AvailBandwidth=0″
     BundleId=4,Delay=10,Cost=10,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=192,AvailBandwidth=96,
     AtMaxRate=STS48,AvailBandwidth=2,
     BundleId=15,Delay=20,Cost=5,SRLG=0&0&0&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS1,TotalBandwidth=384,AvailBandwidth=336,
     AtMaxRate=STS192,AvailBandwidth=1″
;
```

Example NTDR output for the **RTRV-NTDR** command with ALL:

```
RTRV-NTDR:LT-ONNS-61::123456::ALL;

   LT-ONNS-61 2002-07-14 09:45:28
M  123456 RTRV
   ″NTDR,ALL,135.13.44.61,NumberOfLSAs=3″
   ″LSAAdvertisedRouter=135.13.44.61,LinkCount=3,LT-ONNS-61,
UIipAddr=135.3.9.61,SIGipAddr=135.22.19.55:
   L1Routing=PRIMARY,RoutingAreaID=111.234.215.6,ParentRoutingAreaID=
111.214.115.7,L1RoutingControllerID=218.3.45.56,L1RoutingControllerIPAddr=
104.34.49.1,AbstractModel=ABSTRACTNODE,AbstractNode=129.239.4.23:
   LinkId=135.13.44.61&135.13.3.111,BundleCount=1:
     BundleId=4,Delay=10,Cost=10,SRLG=10&26&322&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=96,AvailBandwidth=96,
     AtMaxRate=STS48,AvailBandwidth=2,
```

```
   LinkId=135.13.44.61&135.13.3.113,BundleCount=2:
     BundleId=1,Delay=10,Cost=6,SRLG=101&29&3272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48,AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
     BundleId=2,Delay=10,Cost=4,SRLG=20&69&272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48,AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
   LinkId=135.13.44.61&135.13.3.21,BundleCount=1:
     BundleId=3,Delay=10,Cost=1,SRLG=30&9&2272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48,AvailBandwidth=0,
     AtMaxRate=STS48,AvailBandwidth=0"
>
   LT-ONNS-61   2002-01-14   09:45:29
M  123456 RTRV
   "LSAAdvertisedRouter=135.13.3.21,LinkCount=3,LT-ONNS-221,
UIipAddr=135.3.9.21,SIGipAddr=135.22.19.56::
   LinkId=135.13.3.21&135.13.144.221,BundleCount=1:
     BundleId=23,Delay=10,Cost=10,SRLG=40&2&472&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=96,AvailBandwidth=96,
     AtMaxRate=STS48,AvailBandwidth=2,
   LinkId=135.13.3.21&135.13.14.216,BundleCount=1:
     BundleId=1,Delay=10,Cost=12,SRLG=50&39&72&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=96,AvailBandwidth=96,
     AtMaxRate=STS48,AvailBandwidth=2,
   LinkId=135.13.3.21&135.13.44.61 ,BundleCount=1:
     BundleId=35,Delay=10,Cost=1,SRLG=30&9&2272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48,AvailBandwidth=0,
     AtMaxRate=STS48,AvailBandwidth=0"
>    ....
   LT-ONNS-61   2002-01-14   09:45:30
M  123456 COMPLD
   "LSAAdvertisedRouter=135.13.3.113,LinkCount=2,LT-ONNS-13,
UIipAddr=135.3.9.13,SIGipAddr=135.22.19.57::
   LinkId=135.13.3.113&135.13.44.61,BundleCount=2:
     BundleId=31,Delay=10,Cost=6,SRLG=101&29&3272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48, AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
     BundleId=32,Delay=10,Cost=4,SRLG=20&69&272&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=48,AvailBandwidth=48,
     AtMaxRate=STS48,AvailBandwidth=1,
   LinkId=135.13.3.113&135.13.144.221,BundleCount=1:
     BundleId=33,Delay=10,Cost=10,SRLG=40&2&472&0&0&0&0&0&0&0,IncludeID=0,
     AtMinRate=STS1,TotalBandwidth=96,AvailBandwidth=96,
     AtMaxRate=STS48,AvailBandwidth=2"
;
```

Example output for the **RTRV-NTDR** command by LSAID:

```
RTRV-NTDR:LT-ONNS-165::123456::LSAID,135.13.44.61;
```

```
   LT-ONNS-165 2001-01-14 09:45:28
M  123456 COMPLD
   ″NTDR,LSAID=135.13.44.61,135.13.44.165,NumberOfLSAs=1″
   ″LSAAdvertisedRouter=135.13.44.61,LinkCount=3,LT-ONNS-61,
UIipAddr=135.3.9.61,SIGipAddr=135.25.18.77:
             L1Routing=PRIMARY,RoutingAreaID=111.234.215.6,ParentRoutingAreaID=
111.214.115.7,L1RoutingControllerID=218.3.45.56,L1RoutingControllerIPAddr=
104.34.49.1,AbstractModel=ABSTRACTNODE,AbstractNode=129.239.4.23:
   LinkId=135.13.44.61&135.13.3.111,BundleCount=1:
     BundleId=4,Delay=10,Cost=10,SRLG=10&26&322&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS48,TotalBandwidth=2,AvailBandwidth=2,
     AtMaxRate=STS48,AvailBandwidth=2,

   LinkId=135.13.44.61&135.13.3.113,BundleCount=2:
     BundleId=1,Delay=10,Cost=6,SRLG=101&29&3272&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS48,TotalBandwidth=1,AvailBandwidth=1,
     AtMaxRate=STS48,AvailBandwidth=1,
     BundleId=2,Delay=10,Cost=4,SRLG=20&69&272&0&0&0&0&0&0&0,IncludeId=0,
     AtMinRate=STS48,TotalBandwidth=1,AvailBandwidth=1,
     AtMaxRate=STS48,AvailBandwidth=1,

   LinkId=135.13.44.61&135.13.3.21,BundleCount=1:
     BundleId=3,Delay=10,Cost=1,SRLG=SRLG=30&9&2272&0&0&0&0&0&0&0,
IncludeId=0,
     AtMinRate=STS48,TotalBandwidth=1,AvailBandwidth=0,
     AtMaxRate=STS48,AvailBandwidth=0″

;
```

Newlines between LinkID's are optional.

Error responses

Validation rule: Invalid componentID parameter II

The command is rejected when for the *datatype* = LSAID the received *ASTNNodeID* is invalid.

Error code: `E511 – IINO – Input, Invalid NOde_ID`

Error code: `E512 – IINI Error – Input, Invalid Neighbor node ID`

Validation rule: Invalid componentID parameter II

The following list identifies conditions that will be returned, if access to the topology database is not possible due to internal problems:

Error code:

`E002 – SICE Error – Status, Inter-Processor Communication Error caused cmd failure`

Error code:

`E003 – SNTT Error – Status, Network Topology database Timed out`

Related TL1 messages

- “ENT-PATH: Create ONNS path connection” (p. 3-1701)
- “ED-PATH: Add or modify alias of ONNS connection” (p. 3-739)
- “DLT-PATH: Delete ONNS path connection” (p. 3-222)
- “REPT-PTHCHG notification” (p. 3-2149)
- “REPT-PTHEVT notification” (p. 3-2175)
- “ENT-CONN: Establish network connections with TNA address parameters” (p. 3-1455)
- “DLT-CONN: Delete network connection” (p. 3-139)
- “REPT CONNCHG notification” (p. 3-2104)
- “ACT-L1ROUTING” (p. 3-15)
- “CANC-L1ROUTING” (p. 3-54)
- “REPT L1ROUTING notification” (p. 3-2133)

Possible issues when widely using unidirectional ONNS connections

When in an ONNS network unidirectional ONNS connections are widely used, the following misbehavior may occur:

1. Local audit erroneously reports extra NN cross-connections: **AUD-LOCAL** and **RTRV-AUDLOG** may erroneously report extra NN cross-connections, if two unidirectional ONNS connections of differing bandwidth are traversing a node in opposite directions, but with connecting the same timeslots, for example, one connection from timeslot A to timeslot B and the other one from timeslot B to timeslot A.
2. Free bandwidth generally advertised on an INNI link, although no more bandwidth for bidirectional ONNS connections available: Setup of a bidirectional ONNS connection via an INNI link may fail, although for this link enough free bandwidth is generally advertised for example, via **RTRV-NTDR**. This is the case, if the corresponding link is nearly totally filled with a major part being unidirectional connections. The advertised free bandwidth may then refer to differing timeslots for both directions, thus not allowing bidirectional connections to be setup.

Incorrect bandwidth reporting

In case a *unidirectional* Signal Degrade (SD) condition occurs on an INNI port connection within an ONNS domain, this SD condition leads to a near-end-failure condition being raised for the affected port connection by the detecting node for the corresponding direction only. This leads to the following behavior:

- For the direction opposite to the direction of the corresponding SD condition the bandwidth of the affected port connection is removed of the available bandwidth advertised for the corresponding INNI link bundle, although in this direction no failure condition at all is present.
- Asymmetric available bandwidth is reported for the affected INNI link bundle via **RTRV-NTDR**, but just the other way round as the actual failure condition would suggest.
- For the setup of a unidirectional ONNS path (due to connection setup or restoration) via the affected link bundle in the direction opposite to the SD condition the affected port connection will not be selected, although in this direction no failure condition exists and the setup would succeed.
- The setup of a unidirectional path in the direction of the SD condition or of a bidirectional path in any of both directions via the affected port connection would be possible and would succeed in first place, but might afterwards still fail with E022 due to the fact, that the SD condition on port connection level induced an SD condition on path level.
  This is due to the fact, that for path level an SD condition raises a failure condition that causes a path setup to fail.

Workaround

In case such asymmetric behavior occurs, INNI link bundles in the affected ONNS domain could be checked for SD conditions via **RTRV-NTDR**, **RTRV-NNDR**, and **RTRV-rr**. Additionally, an SD alarm should have been raised for the affected port, if not suppressed.

RTRV-OC3: Retrieve OC3 port

The command **RTRV-OC3** is used to retrieve all parameters related to the optical OC3 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-OC3**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC3 port.
   Type: AID.
   Range: SPortAID of OC3 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-OC3** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID.
   Range: OC3 port AID.

Spec block parameters

1. **adminCost**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **brstintvl**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **brstthr** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the Far-end Line.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is setable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
    Default: 31.
    Example: An O-3 signal might be represented by 31 which implies 3 STS1's or 13 which implies 1 STS3C.
11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONET.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
15. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 or STS3 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default) or STS3.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
   - IPV4 (default)
   - IPV6
   - NSAP

26. **`networkTNAAddress`** – Network TNA Address

    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

    This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

    This parameter is supported by ports of *portClass* UNI.

    This parameter is supported by ports of *portClass* EDGE.

    Type:

    - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

    Range:

    - IPV4: Each numeric string: 0…255
    - IPV6: Each hexadecimal number: 0000…FFFF
    - NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    Examples:

    - IPV4: 129.12.3.178
    - IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    - NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

    Default:

    - IPV4: 0.0.0.0
    - IPV6: 0-0-0-0-0-0-0-0
    - NSAP: 0000000000000000000000000000000000000000

    The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`opdis`** – Optical Distance

    It shows the type of application for which the optical transmitter on this port is suited.

    Type: Alphanumeric value set.

Range:

- VERY-SHORT-REACH
- SHORT-REACH
- INTERMEDIATE-REACH
- LONG-REACH
- VERY-LONG-REACH
- OLS_PASSIVE
- OLS_400G
- OLS_800G
- XTREME.
- NA.

28. **`opwl`** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM) NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

29. **`outql`** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SONET line overhead byte.
Type: Alphanumeric value set.
Range:
    - PRS
    - STU
    - ST2
    - ST3
    - DUS (default).

30. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-1/OC-3: 1 – 192000.

Default: The default values differ for SDH and SONET:

- STM1: 28800
- OC3: 154.

31. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

32. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

33. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000

    Default: The default values differ for SDH and SONET and the different rates:

    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.

34. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
35. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
36. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0 (default) …25. The unit of this value is ″milliseconds″.
37. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: OC3.
38. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

39. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
40. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
41. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.

43. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST
    - POISSON (default).
45. **`sdthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
46. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`sfthr`** – Signal Failure Threshold
Signal failureb threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
48. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

49. **`sonet`** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see "RTRV-STM4: Retrieve STM4 port" (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen ("-") is allowed in the name additionally.
Default: DEFAULT.

50. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

51. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter

strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

53. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
- 1: for specific repeating byte (SONET)
- 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

54. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

55. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.

56. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
57. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899) for SDH equivalent.
    Type: Alphanumeric value set.
    Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
58. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1 or 3 representing STS1 or STS3C.
    Default: 3 1 = all STS1.
59. **`z0outmode`** – Mode of outgoing Z0 byte insertion
    The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
    Type: Alphanumeric value set.
    Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-OC3** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-OC3** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-OC3:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=OC3,frcdus=DISABLE,sfthr=-3,sdthr=-6, pmode=MON,.....
      pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-OC3 command.

Check CCB against *ptype* and *istd*

See definition of error response in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899).

References

GR-199-CORE (RTRV-*rr*)

RTRV-OC12: Retrieve OC12 port

The command **RTRV-OC12** is used to retrieve all parameters related to the optical OC12 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-OC12**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC12 port.
   Type: AID.
   Range: SPortAID of OC12 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-OC12** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ”aid[:spec_block]”
       ...
       ...
   ”aid[:spec_block]”
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-12 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **crstat** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
6. **exprstrc** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **felpm** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the Far-end line.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **frcdus** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
    Default: 121.
    Example:
    An OC12 signal might be represented by 31-33 which implies 3 STS1's, 3 STS3C's.

11. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONET.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

12. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

14. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3 or STS12 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3 or STS12.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of portClass = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **`nelpm`** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
    - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

    Range:
    - IPV4: Each numeric string: 0…255
    - IPV6: Each hexadecimal number: 0000…FFFF
    - NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    Examples:
    - IPV4: 129.12.3.178
    - IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    - NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

    Default:
    - IPV4: 0.0.0.0
    - IPV6: 0-0-0-0-0-0-0-0
    - NSAP: 0000000000000000000000000000000000000000

    The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
27. **`opdis`** – Optical Distance
It shows the type of application for which the optical transmitter on this port is suited.
Type: Alphanumeric value set.

Range:

- VERY-SHORT-REACH
- SHORT-REACH
- INTERMEDIATE-REACH
- LONG-REACH
- VERY-LONG-REACH
- OLS_PASSIVE
- OLS_400G
- OLS_800G
- XTREME
- NA.

28. **opwl** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM) NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

29. **outql** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SONET line overhead byte.
Type: Alphanumeric value set.
Range:
   - PRS
   - STU
   - ST2
   - ST3
   - DUS (default).

30. **pmmsfeses** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-4/OC-12: 1 – 768000.

Default: The default values differ for SDH and SONET:

- STM4: 115200
- OC12: 615.

31. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-4/OC-12: 1 – 768000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM4: 115200
    - OC3: 615.

32. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

33. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000

    Default: The default values differ for SDH and SONET and the different rates:

    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.

34. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
35. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
36. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …25. The unit of this value is ″milliseconds″.
37. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: OC12.
38. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

39. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
40. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
41. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
    Default: empty string: ""
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.

43. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
45. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
46. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`sfthr`** – Signal Failure Threshold
    Signal failure threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
48. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.

49. **sonet** – SONET ASAP
SONET Alarm Severity Assignment Profile. This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH ASAP for SDH ports of corresponding line rate (see “RTRV-STM4: Retrieve STM4 port” (p. 3-2881), output parameter *sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string.
Range: String with a maximum of 24 characters. A hyphen (“-”) is allowed in the name additionally.
Default: DEFAULT.

50. **spm** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

51. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **strc** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter

strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

53. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
54. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex
55. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.

56. **strcwfmt** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
57. **trbmd** – Tributary mode
    If *trbmd* is FIXED, then inputsig constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in RTRV-STM16 for SDH equivalent.
    Type: Alphanumeric value set.
    Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400) command.
58. **unequiposig** – Unequipped Output Signal List
    This parameter is a list of STSn rates across the port’s bandwidth. It determines the rate(s) of the “unequipped” signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port are allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3 or 12 representing STS1, STS3C or STS12C.
    Default: 121 = all STS1 (fixed).
59. **z0outmode** – Mode of outgoing Z0 byte insertion
    The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
    Type: Alphanumeric value set.
    Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-OC12** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-OC12** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-OC12:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=OC12,frcdus=DISABLE,sfthr=-3,sdthr=-6, pmode=MON,.....
      pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-OC12 command.

Check CCB against ptype and istd

See definition of error response in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899).

References

GR-199-CORE (RTRV-*rr*)

RTRV-OC48: Retrieve OC48 port

The command **RTRV-OC48** is used to retrieve all parameters related to the optical OC48 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-OC48**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC-48 port.
   Type: AID.
   Range: SPortAID of OC48 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-OC48** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID.
   Range: OC-48 port AID.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48 representing STS1, STS3C, STS12C, STS48C.
6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the Far-end Line.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE , set a forced DUS/DNU
   - DISABLE (default), clear a forced DUS/DNU

9. **fsflpbkstat** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
   - YES
   - NO (default).
10. **inputsig** – Tributary Input Signal Rate List
This parameter is a list of STSn rates across the port's bandwidth.
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
Default: 481.
Example: An OC-48 signal might be represented by 91-13-112-241 which implies 9 STS1's, 1 STS3C, 1 STS12C, 24 STS1's.
11. **istd** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET.
Type: Alphanumeric value set.
Range: SONET.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
12. **linkResCls** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

13. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

14. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
   - YES
   - NO (default).

16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12 or STS48 (default)
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12 or STS48.
18. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

23. **neighborSigPCIPAddr** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

24. **nelpm** – Near-End Line PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

25. **networkTNAType** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

26. **`networkTNAAddress`** – Network TNA Address

    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

    This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

    This parameter is supported by ports of *portClass* UNI.

    This parameter is supported by ports of *portClass* EDGE.

    Type:

    - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

    Range:

    - IPV4: Each numeric string: 0…255
    - IPV6: Each hexadecimal number: 0000…FFFF
    - NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    Examples:

    - IPV4: 129.12.3.178
    - IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    - NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

    Default:

    - IPV4: 0.0.0.0
    - IPV6: 0-0-0-0-0-0-0-0
    - NSAP: 0000000000000000000000000000000000000000

    The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

27. **`opdis`** – Optical Distance

    It shows the type of application for which the optical transmitter on this port is suited.

    Type: Alphanumeric value set.

Range:

- VERY-SHORT-REACH
- SHORT-REACH
- INTERMEDIATE-REACH
- LONG-REACH
- VERY-LONG-REACH
- OLS_PASSIVE
- OLS_400G
- OLS_800G
- XTREME
- NA.

28. **opwl** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM) NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.
29. **outql** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SONET line overhead byte.
Type: Alphanumeric value set.
Range:
    - PRS
    - STU
    - ST2
    - ST3
    - DUS (default).
30. **pmmsfeses** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

31. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

32. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

33. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.

34. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

35. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

36. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

37. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: OC48.
38. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
39. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
40. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

41. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
42. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable consequent action for TIM
    - DISABLE (default): disable consequent action for TIM.
43. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
44. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST
    - POISSON (default).
45. **`sdthr`** – Signal Degrade Threshold
    Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.

46. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
47. **`sfthr`** – Signal Failure Threshold
    Signal failure threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
    Type: numeric value set.
    Range (as exponents of 10): –5; –6; –7; –8; –9.
    Default: –6.
48. **`sltca`** – Section and Line TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
49. **`sonet`** – SONET ASAP
    This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (see “RTRV-STM16: Retrieve STM16 port” (p. 3-2899), output parameter *sdhpn*) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
50. **`spm`** – Section PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
51. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.

If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

53. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

54. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

55. **strcst** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.

56. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

57. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect.
Type: Alphanumeric value set.
Range:
    - ADAPTIVE
    - FIXED.

    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

58. **unequiposig** – Unequipped Output Signal List
This parameter is a list of STSn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed. If *trbmd* is

ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 1, 3, 12 or 48 representing STS1, STS3C, STS12C or STS48C.
Default: 481 = all STS1 (fixed).

59. z0outmode – Mode of outgoing Z0 byte insertion
The Z0 insertion is configurable to support interworking with old SONET equipment for STS-M interfaces (M=3, 12, 48).
Type: Alphanumeric value set.
Range:
    - FIXEDCC (default): for inserting the fixed value CChex
    - STSID: for inserting the STS-Id of the current frame.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-OC48** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-OC48** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-OC48:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ”1-1-#-#-1-1:ptype=OC48,frcdus=DISABLE,sfthr=-3,sdthr=-6, pmode=MON,.....
      pmmsneses=15,pmrsneses=30”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-OC48 command.

Check CCB against ptype and istd

See definition of error response in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899).

References

GR-199-CORE (RTRV-*rr*)

RTRV-OC48T: Retrieve OC48T port

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **`RTRV-OC48T`**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   Access Identifier of an OC-48 port.
   Type: AID.
   Range: SPortAID of OC48 port.
   The keyword **`all`** is allowed up to shelf level.
3. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-OC48T command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of an OC-48 port.
   Type: AID.
   Range: SPortAID of OC48 port.
   The keyword **`all`** is allowed up to shelf level.

Spec block parameters

1. **`exprstrc`** – Value of Expected J0
   For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
3. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).

4. **istd** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET.
   Type: Alphanumeric value set.
   Range: SONET.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
5. **lpbk** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).
6. **opdis** – Optical Distance
   It shows the type of application for which the optical transmitter on this port is suited.
   Type: Alphanumeric value set.

   Range:

   - VERY-SHORT-REACH
   - SHORT-REACH
   - INTERMEDIATE-REACH
   - LONG-REACH
   - VERY-LONG-REACH
   - OLS_PASSIVE
   - OLS_400G
   - OLS_800G
   - XTREME
   - NA.
7. **opwl** – Optical Wavelength
   Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
   Type: Alphanumeric value set.
   Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM) NA (for client ports).
   The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

8. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
9. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
10. **`pmrsneses`**out – Performance Monitoring Section/RS Near-End SES Threshold in Outgoing Direction
   This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set
   Range (superset of allowed ranges for SDH and SONET): 1 – 64000
   Default:
   The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
11. **`ptype`** – Port Type
   This identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range: OC48T.

12. **rstimmon** – RS TIM Monitor Enable
For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
13. **rstimmonout** – RS TIM Monitor Enable in Outgoing direction
Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor
14. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
15. **sltca** – Section and Line TCA Profile Pointer
Definition of the section and line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
16. **sonet** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (*sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
17. **spm** – Section PM Enable
For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
Enable or disable section performance monitoring.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

18. **`spmout`** – Section PM Enable in Outgoing Direction
Enable or disable section performance monitoring for the outgoing (egress) direction.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).

19. **`strc`** – Value of incoming J0
For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

20. **`strcfmt`** – Format of incoming J0
For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).

21. **`strcfmtout`** – Format of outgoing J0
The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
Type: Numeric string.
Range:
   - 1, for specific repeating byte (SONET)
   - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)

- 16, for 16 byte specific string (ITU/ETSI mode 1)
- 64, for 64 byte string (SONET).

22. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
23. **`strcst`** – Format of incoming J0
    For RTRV-OC48T this parameter refers to the incoming (ingress) direction.
    This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
    Type: Alphanumeric value set.
    Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
24. **`strcstout`** – Status of outgoing J0
    This parameter shows the status of the value of the RS/Section Trace Identifier in outgoing (egress) direction which is available with parameter strcout.
    alphanumeric value set
    Range:
    - NORMAL, strcout contains a valid value
    - UNAVAILABLE, no value for strcout could be obtained by the system
    - AIS, no value for strcout due to an SF condition

Detailed behavior description

The command retrieves the parameters as described above.

The **`RTRV-OC48T`** command does not generate a **`REPT DBCHG`** message.

Example input/output

The following example shows the successful execution of a **RTRV-OC48T** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-OC48T:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=OC48T,pmode=MON,..... pmrsneses=30,pmrsnesesout=30″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-OC48T** command.

Check CCB against *ptype* and *istd*

See definition of error response in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899).

References

GR-199-CORE (RTRV-*rr*)

RTRV-OC192: Retrieve OC192 port

The command **RTRV-OC192** is used to retrieve all parameters related to the optical OC192 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-OC192**:*tid:aid:ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an OC192 port.
   Type: AID.
   Range: SPortAID of OC192 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-OC192** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier of the OC192 port.
   Type: AID.
   Range: SPortAID of OC192 port.
   The keyword **`all`** is not allowed.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future.
This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. It is the pointer processor setup if adaptive mode (*trbmd* = ADAPTIVE) is selected.
   When switching *trbmd* from FIXED to ADAPTIVE or when creating a port newly *crstat* inherits the current value of *inputsig* until the system delivers an up-to-date value from the pointer processor. If no value is available from the pointer processor the latest known value is retained.
   For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the tributary input rate signal list.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C, STS192C.
6. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
7. **`fec`** – FEC Enable
   Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
   Type: Alphanumeric value set.
   Range:
   DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics
   - OC192 / STM64 client ports (related to OTN ports)
   - OC768/STM256 basis low cost intra office 1310 nm optics.

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

ENABLE (only) for:

- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G

Default:

DISABLE for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768/STM256 basis low cost intra office 1310 nm optics.

ENABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics with strong FEC (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

8. **`felpm`** – Far-end Line PM Enable
   Enable or disable of the performance monitoring of the Far-end Line.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **`frcdus`** – Force to DUS
   Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
   Type: Alphanumeric value set.
   Range:
   - ENABLE: (set a forced DUS/DNU)
   - DISABLE (default): (clear a forced DUS/DNU).

10. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

11. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of STSn rates across the port's bandwidth.
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then *inputsig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3, 12, 48, 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
    Default: 1921.

12. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET.
    Type: Alphanumeric value set.
    Range: SONET.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

13. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

14. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
15. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
16. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

17. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1, STS3, STS12, STS48 or STS192 (default)
18. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: STS1 (default), STS3, STS12, STS48 or STS192.
19. **`msrei`** – MS-REI mode
    An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
    Type: Alphanumeric value set.
    Range: SINGLE or EXTENDED (default).
20. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborASTNNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.
22. **`neighborNodeID`** – Neighbor Node Id
    This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborNodeID is used by the ONNS application.
    Type: Four dot separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0.0.0.0
    If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″
23. **`neighborPortID`** – Neighbor Port Id
    This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
    If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
    neighborPortID is used by the application.
    Type: Four dash separated numeric string.
    Range: Each numeric string: 0…255.
    Default: 0-0-0-0.
    Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
    The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
26. **`nelpm`** – Near-End Line PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
27. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.

Range:

- IPV4 (default)
- IPV6
- NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`och`** – optical channel enable
    Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don't have any impact on transmission.
    Type: Alphanumeric value set.
    Range:
    - for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
    - for ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
    - for client ports on packs which support OTN: DISABLE: disable, optical channel not applicable.

    Default:
    - for OP10 and OP40 packs, which cannot support OCH: DISABLE
    - for OP10 packs, which can support OCH: ENABLE
    - or OP10 and OP40 packs, which only support OCH: ENABLE
    - for client ports on packs which support OTN: DISABLE.
30. **`opdis`** – Optical Distance
    It shows the type of application for which the optical transmitter on this port is suited.
    Type: Alphanumeric value set.
    Range:
    - VERY-SHORT-REACH
    - SHORT-REACH
    - INTERMEDIATE-REACH
    - LONG-REACH
    - VERY-LONG-REACH
    - OLS_PASSIVE
    - OLS_400G
    - OLS_800G
    - XTREME
    - NA.

31. **`opwl`** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM) NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.
32. **`otnclient`** – OTN Client Port
This parameter indicates whether the port is an OTN Client Port or an optical SDH/SONET port.
Type: Alphanumeric value set.
Range: YES for OTN client port. NO for optical SDH/SONET port.
33. **`outql`** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SONET line overhead byte.
Type: Alphanumeric value set.
Range:
   - PRS
   - STU
   - ST2
   - ST3
   - DUS (default).
34. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-16/OC-48: 1 – 3072000
   - STM-64/OC-192: 1 – 12288000
   - STM-256/OC-768: 1 – 49152000.

   Default: The default values differ for SDH and SONET and the different rates:
   - STM16: 460800
   - STM64: 1843200
   - STM256: 7372800
   - SONET OC48: 2459

- SONET OC192: 9835
- SONET OC768: 39340.

35. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.
36. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start. The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

    Default:
    - AUTO for optical SDH/SONET ports
    - NMON for client ports.

37. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
38. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (och = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.
39. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
40. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
41. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

42. **`ptype`** – Port Type
    This identifies the port type of the AID.
    It is derived from the functional name and functional qualifier of a circuit pack. If the functional qualifier indicates a pack with 10GE_WANPHY interfaces the port type has the value 10GE_WANPHY, else OC192.
    Type: Alphanumeric value set.
    Range: OC192.
43. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
44. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
45. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

46. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

47. **rstimca** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
   - ENABLE: enable consequent action for TIM
   - DISABLE (default): disable consequent action for TIM.

48. **rstimmon** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
   - ENABLE: enable TIM Monitor
   - DISABLE (default): disable TIM Monitor

49. **sdsfmode** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
   - BURST
   - POISSON (default).

50. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

51. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
52. **`sfthr`** – Signal Failure Threshold
Signal failure threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
53. **`sltca`** – Section and Line TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
54. **`sonet`** – SONET ASAP
This parameter sets a SONET ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SDH Alarm Severity Assignment Profile for SDH ports (*sdhpn*) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
55. **`spm`** – Section PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
56. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are

set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

57. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

58. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).

59. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

60. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.

61. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

62. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* and *unequiposig* have no effect. See definition of output parameter *trbmd* in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899) for SDH equivalent.
Type: Alphanumeric value set.

Range:

- ADAPTIVE
- FIXED.

Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in “RTRV-EQPT (system): Retrieve Equipment (system)” (p. 3-2400) command.

63. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of STSn rates across the port’s bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need to have the identical setting. Mixed rates for tx and rx side of the port is allowed. If *trbmd* is ADAPTIVE, then *unequiposig* is settable and retrievable as in fixed mode but does not influence the port behavior.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 1, 3, 12, 48 or 192 representing STS1, STS3C, STS12C, STS48C or STS192C.
    Default: 1921 = all STS1 (fixed).

Detailed behavior description

The command retrieves the parameters as described above.

The command **`RTRV-OC192`** does not generate a **`REPT DBCHG`** message.

Example input/output

The following example shows the successful execution of a RTRV-OC192 command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-OC192:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=OC192,frcdus=DISABLE,sfthr=-3,sdthr=-6,emi=TRUE,
      pmode=MON,..... pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-OC192 command.

Check CCB against ptype and istd

See definition of error response in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899).

References

GR-199-CORE (RTRV-*rr*)

RTRV-OTU2: Retrieve OTU2 port

The command **`RTRV-OTU2`** is used to retrieve all parameters related to the OTU2 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **`RTRV-OTU2`**:*tid:aid:ctag*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   Access Identifier of an OTU2 port.
   Type: AID
   Range: OTNPortAID of OTU2 port.
   The keyword ″otall″ is allowed at port level, **`all`** up to shelf level.
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string.
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831

Output format

In response to a valid RTRV-OTU2 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   Type: AID.
   Range: OTNPort AID.
   The AID value ″otall″ is not allowed in the response.

Spec block parameters

1. **expotutti** – Expected OTU TTI
   Expected OTU trail trace identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the expected OTU TTI.
   Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
   Range: 15 byte hexadecimal string
   Default: 15 bytes each with value 00.
2. **fectype** – FEC Type
   Forward error correction type.
   Type: Alphanumeric value set.
   Range: RS_FEC, V_FEC (default).
3. **ochpm** – OCH PM Enable
   Enable or disable optical channel performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE (default).
4. **opwind** – Optical Window
   Optical window (L-Band, C-Band) supported by the port.
   Type: Alphanumeric value set.
   Range: BAND_L for L-Band, or BAND_C for C-Band.
5. **opwl** – Optical Wavelength
   Optical Wavelength in nanometers.
   Type: numeric value set
   Range:
   - 35 wavelengths in C-Band (1554.537 … 1568.362), i.e:
     1554.537, 1554.940, 1555.343, 1555.747, 1556.151, 1556.555, 1556.959, 1557.363, 1557.768, 1558.173, 1558.578, 1558.983, 1559.389, 1559.794, 1560.200, 1560.606, 1561.013, 1561.419, 1561.828, 1562.233, 1562.640, 1563.047, 1563.455, 1563.863, 1564.271, 1564.679, 1565.087, 1565.496, 1565.905, 1566.314, 1566.723, 1567.133, 1567.542, 1567.952, 1568.362
   - 93 wavelengths in L-Band (1568.773 … 1607.466), i.e:

1568.773, 1569.183, 1569.594, 1570.005, 1570.416, 1570.828, 1571.239, 1571.651, 1572.063, 1572.476, 1572.888, 1573.301, 1573.714, 1574.127, 1574.540, 1574.954, 1575.368, 1575.782, 1576.196, 1576.610, 1577.025, 1577.440, 1577.855, 1578.270, 1578.686, 1579.102, 1579.518, 1579.934, 1580.350, 1580.767, 1581.184, 1581.601, 1582.018, 1582.436, 1582.854, 1583.271, 1583.690, 1584.108, 1584.527, 1584.946, 1585.365, 1585.784, 1586.203, 1586.623, 1587.043, 1587.463, 1587.884, 1588.304, 1588.725, 1589.146, 1589.568, 1589.989, 1590.411, 1590.833, 1591.255, 1591.678, 1592.100, 1592.523, 1592.946, 1593.369, 1593.793, 1594.217, 1594.641, 1595.065, 1595.489, 1595.914, 1596.339, 1596.764, 1597.189, 1597.615, 1598.041, 1598.467, 1598.893, 1599.320, 1599.746, 1600.173, 1600.600, 1601.028, 1601.455, 1601.883, 1602.311, 1602.740, 1603.168, 1603.597, 1604.026, 1604.455, 1604.885, 1605.314, 1605.744, 1606.174, 1606.605, 1607.035, 1607.466

- Value 0 (default) (i.e. undefined).

6. **`otnasap`** – OTN ASAP
   OTN Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type OTN to which the designated AID points.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: DEFAULT.
7. **`otutimca`** – OTU TIM Consequent Action Enable
   OTU Trail Trace Identifier Mismatch consequent action enable. Enabling or disabling the the TIM consequent action shall also enable or disable the TIM detection.
   Type: Alphanumeric value set.
   Range: ENABLE: enable consequent action for TIM (default), DISABLE: disable consequent action for TIM.
8. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default):
     indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON:
     indicates that alarm monitoring is taking place on the port.
   - NMON:
     means that alarm monitoring is not occurring on the port.
9. **`ptype`** – Port Type
   This identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range: OTU2

10. **`strecotutti`** – Status of incoming OTU TTI
    This parameter shows the status of the value of the OTU Trail Trace Identifier which is available with parameter voturectti.
    Type: Alphanumeric value set.
    Range:
    - NORMAL: voturectti contains a valid value
    - UNAVAILABLE: no value for voturectti could be obtained by the system
    - AIS: no value for voturectti due to an SF condition.
11. **`txenable`** – Optical Output Enable
    Optical output enabling.
    Type: Alphanumeric value set.
    Range:
    - DISABLE (default): optical output disabled
    - ENABLE: optical output enabled.
12. **`voturectti`** – Value of incoming OTU TTI
    Value of the received OTU Trail Trace Identifier. The status of this value is contained in parameter strecotutti. If strecotutti indicates NORMAL, then voturectti contains a valid value. If strecotutti indicates UNAVAILABLE or AIS, then no incoming OTU TTI could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter voturectti is omitted from the response. The parameter indicates the Source Point Access Identifier (SAPI) of the received OTU TTI.
    Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
    Range: 15 byte SAPI (Source Access Point Identifier).
13. **`votutti`** – Value of outgoing OTU TTI
    Value of the transmitted OTU Trail Trace Identifier. The parameter indicates the Source Point Access Identifier (SAPI) of the transmitted OTU TTI.
    Type: Hexadecimal string (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
    Range: 15 byte SAPI (Source Access Point Identifier).
    Default: 15 bytes each with value 00.

Detailed behavior description

The command retrieves the parameters as described above. The command **`RTRV-OTU2`** does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful execution of a RTRV-OTU2 command:

```
RTRV-OTU2:LT-UNITE:1-1-#-#-1-ot1:123456;

   LT-UNITE 06-06-30 08:00:00
```

```
M  123456 COMPLD
   ″1-1-#-#-1-ot1:expotutti=4C5543454E544E45312D504F525431,fectype=V_
      FEC,ochpm=ENABLE,opwind=BAND_
      L,opwl=1569.183,otuasap=DEFAULT,otutimca=DISABLE,pmode=MON,ptype=OTU2,strecotutti=NORM
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-OTU2 command.

Related TL1 messages

- “ED-OTU2: Edit OTU2 port” (p. 3-730)

References

GR-199-CORE (RTRV-*rr*)

RTRV-OUC: Retrieve ONNS to UPSR/SNCP constructs

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-OUC:*tid:wrk_aid:ctag[::ouc_dir];*

Input parameters

Command parameters

1. **tid** – Target Identifier
See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **wrk_aid** – Working Tributary Access Identifier
This parameter specifies the working tributary of the OUC.
Type: STS1NumAID
Range:
   - The *1-1-#-#-all-all-all*-value is accepted in general for this parameter.
   - The *all* value is not accepted, if specified only for the slot-, port- or trib- part of this parameter.
   - The *1-1-#-#-all-all-all*-value is not accepted, if the parameter ouc_dir is present.
3. **ctag** – Correlation Tag
This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
Type: Alphanumeric string
Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **ouc_dir** – OUC Directionality
Directionality of the OUC
Type: Alphanumeric value set.
Range:
   - BIDIR
   - UNIDIR-ADD
   - UNIDIR-DROP.

Output format

In response to a valid RTRV-OUC command, the following output is returned:

```
   sid date time
M  ctag COMPLD
   "wrk_aid,prt_aid:ouc_dir,ouc_rate"
;
or in case of more than one OUC retrieved:


   sid date time
M  ctag COMPLD
   "wrk_aid,prt_aid:ouc_dir,ouc_rate"
       ...
       ...
   "wrk_aid,prt_aid:ouc_dir,ouc_rate"
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`wrk_aid`** – Working Tributary Access Identifier
   This parameter specifies the working tributary of the OUC.
   Type: STS1NumAID
   Range: n.a. (*all* values shall never occur as part of an output parameter)
2. **`prt_aid`** – Protection Tributary Access Identifier
   This parameter specifies the protection tributary of the OUC.
   Type: STSNumAID
   Range: n.a. (*all* values shall never occur as part of an output parameter)
3. **`ouc_dir`** – OUC Directionality
   Directionality of the OUC
   Type: Alphanumeric value set.
   Range:
   - BIDIR (default)
   - UNIDIR-ADD
   - UNIDIR-DROP

4. **`ouc_rate`** – OUC Rate
   The parameter *ouc_rate* indicates the rate of the OUC, which is determined by the cross-connection rate of the associated cross-connections. For a detailed description please refer to parameter rate in RTRV-CRS.
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192

Example input/output

Retrieval of the single bidirectional OUC with working tributary 1-1-#-#-1-1-4:

```
RTRV-OUC:UNITE-NE:1-1-#-#-1-1-4:123456::BIDIR;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
   "1-1-#-#-1-1-4,1-1-#-#-2-2-12:BIDIR,STS1"
;
```

Retrieval of all OUCs in an NE:

```
RTRV-OUC:UNITE-NE:1-1-#-#-all-all-all:123456;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
   "1-1-#-#-1-1-4,1-1-#-#-2-2-4:BIDIR,VC4"
   "1-1-#-#-1-1-7,1-1-#-#-2-4-7:UNIDIR-DROP,VC3"
   "1-1-#-#-1-1-10,1-1-#-#-2-2-10:UNIDIR-ADD,VC4"
;
```

Error responses

Validation rule: Invalid wildcarding

OUC command processing shall verify that no wildcard is applied for *wrk_aid* on slot, port or trib level or if *ouc_dir* is present.

Error code: `IDNV – Input, Data Not Valid, Invalid Wildcarding for wrk_aid`

RTRV-OW: Retrieve orderwire and user bytes

General

The **RTRV-OW** command can be initiated by a user to retrieve the overhead access ports and any associated optical port (OP***n***) in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-OW**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   The parameter aid (Access Identifier) specifies the overhead access port.
   Type: UserBytePortAID.
   Range: As given by the *UserBytePortAID* definition, the value ″ubioall″ is supported.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Output format

In response to a valid RTRV-OW command, the following output is returned:

```
   sid date time
M  ctag COMPLD
   ″aid:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the overhead access port for which the following spec_block parameters are returned.
   Type: UserBytePortAID.
   Range: As given by the UserBytePortAID definition, the value ″ubioall″ is not allowed.

Spec block parameters

1. **intfctype** – Interface Type
   This parameter specifies which electrical interface standard of the overhead access port is used.

   Two different settings are possible:

   - electrical interface according to G.703
   - electrical interface according to V.11.

   Note: The overhead access ports with AIDs ubio1, .. ubio4 are configurable to either G703 or V11. The overhead access ports with AIDs ubio5, ubio6 are fixed to V11.
   Type: Alphanumeric value set.
   Range: G703, V11.
   Default: G703 (for the overhead access ports given by the AIDs ubio1 .. ubio4); V11 (for the overhead access ports given by the AIDs ubio5, ubio6 – fixed to V11).
2. **ohbyte** – Overhead-Byte
   This parameter specifies which of the overhead bytes is used.

   Three channels are possible:

   - E1 and F1 in the RSOH (section overhead)
   - E2 in the MSOH (line overhead).

   Type: Alphanumeric value set.
   Range: E1, F1, E2.
   Default: E1.
   Note: In output direction the empty string appears as \″\″.
3. **portaid**
   This parameter specifies the optical port (of an OP***n***) with which the overhead access port (given by the output parameter aid) is associated.
   The parameter value NOCONNECT is used to indicate that the port given by portaid is not associated to the overhead access port given by the parameter aid.

Type: In case an association is established: SPortAID; In case an association is no established: Alphanumeric value set (i.e. the string ″NOCONNECT″).
Range: In case an association is established: SPortAID (No **all** value is supported); in case an association is no established: NOCONNECT.
Default: NOCONNECT.

Detailed behavior description

The **RTRV-OW** command retrieves the overhead access ports and any associated optical port (OP***n***) in the NE.

Example input/output

Example 1

Suppose the UserByte port ubio6 is associated with the port-4 of an OP2G5 in slot03 (the intfctype is fixed to V11 for ubio5 and ubio6):

```
RTRV-OW:UNITE-NE:1-1-ubio6:123456

   sid date time
M  ctag COMPLD
   ″1-1-ubio6:portaid=1-1-#-#-3-4,ohbyte=E1,intfctype=V11″
;
```

Example 2

Suppose the

- UserByte port ubio1 (V11,E1) is associated with the port-4 of an OP2G5 in slot03
- UserByte port ubio3 (G703,E2) is associated with the port-1 of an OP10G in slot07
- UserByte port ubio6 (V11,F1) is associated with the port-4 of an OP10G in slot34
- The other UserByte ports (ubio2, ubio4, ob05) are not associated:

```
RTRV-OW:UNITE-NE:1-1-ubioall:123456

   sid date time
M  ctag COMPLD
   ″1-1-ubio1:portaid=1-1-#-#-3-4,ohbyte=E1,intfctype=V11″
   ″1-1-ubio2:portaid=NOCONNECT,ohbyte=E1,intfctype=G703″
   ″1-1-ubio3:portaid=1-1-#-#-7-1,ohbyte=E2,intfctype=G703″
   ″1-1-ubio4:portaid=NOCONNECT,ohbyte=E1,intfctype=G703″
   ″1-1-ubio5:portaid=NOCONNECT,ohbyte=E1,intfctype=V11″
   ″1-1-ubio6:portaid=1-1-#-#-34-4,ohbyte=F1,intfctype=V11″
;
```

Related TL1 messages

- "SET-OW: Set orderwire and user bytes" (p. 3-3147)

RTRV-PATH: Retrieve ONNS connection data

Overview

The **RTRV-PATH** command requests information on an ONNS path which begins in the source node to which the message was addressed. The ONNS source node determines the status of the connection from the information stored in the Network Navigator databases.

The query must be directed to the source node. Querying an existing ONNS path by its *ConnectionID* or *ConnAlias* alone will return the status on all paths associated with the *ConnectionID*. The query response includes a status message like Connected and OK (CON&OK), the list of network element nodes that the path utilizes, and the list of constraint parameters for the connection.

ONNS supports querying all connections by sending RTRV-PATH with *connID* = ALL When all of the records of the connection data repository are retrieved by RTRV-PATH with *connID* = ALL, each connection record will be the same format as the RTRV-PATH ConnID response message format.

ONNS supports querying all connections of a connection group, i.e. all connections starting at VCGTRIBs of one single VCG, by sending RTRV-PATH with the ALL qualifier for the tributary number within the *connID*. When several records of the connection data repository are retrieved by RTRV-PATH with qualifier ALL for the tributary number in *connID*, each connection record will be the same format as the standard RTRV-PATH *ConnID* response message format. The *connAlias* will not be accepted as input to retrieve the connections of a group and will be ignored, if set.

ONNS supports querying all connections of an M:N protection group, i.e. working as well as protecting connections, by sending RTRV-PATH with the M:N protection group ID within the *connID*. In this case each connection record will be in the same format as the standard RTRV-PATH ConnID response message format.

The connection records will be sorted by their route membership. The connection records of each route will be headlined by a route indicator, the number of connections per route and an group asymmetry indicator.

Additionally querying of single working or protecting connections of an M:N protection group is supported by sending RTRV-PATH with the ID of the corresponding connection within the *connID*.

RTRV-PATH RetrieveDetail_Route parameters

The **RTRV-PATH** command supports the RetrieveDetail_Route function. RetrieveDetail_Route is only available when a single connection ID (connID) is requested. The RetrieveDetail_Route function retrieves data from the network.

RetrieveDetail_Route data is not be stored in the ONNS source node or any other node. RetrieveDetail_Route only operates on stable ONNS connections, with no other ONNS operations pending (including pending RetrieveDetail_Route operations).

RetrieveDetail_Route request is sent to the source node for the specific connection. The message data from individual nodes on the nodelist of the path that is traced is returned to the source node. Each node shall send the node ID, portID, and the timeslot (trib), for each connPoint involved in the path connection specified by pathID. The ability to trace the path though an intermediate node that has had its path cross-connection deleted is supported.

The message data from path nodes is collected, sort by nodeList order, organized by path, and sent as a second response block to the **`RTRV-PATH`** command. The initial RTRV-PATH response block contains the current connection data from the ONNS connection database, uses the RTRV completion code, and is terminated with a ″>″. The second RTRV-PATH response block has the same header line and is terminated with a ″;″.

The ONNS process that controls and receives the RetrieveDetail_Route signaling request responses waits for up to 15 seconds to accumulate the signaling request response messages and then send the data received via TL1 as the second and concluding response block to the **`RTRV-PATH`** command. The RetrieveDetail_Route response indicates any nodes that did not return path data by an node ID entry with no data, and indicates missing cross-connections as needed.

RTRV-PATH RetrieveDetail_State parameters

The **`RTRV-PATH`** command supports the RetrieveDetail_State function. RetrieveDetail_State is only available when a single connection ID (connID) is requested. The RetrieveDetail_State function retrieves connection data (as retrieved by RetrieveDetail_Route) and additionally connection signal states from the network. RetrieveDetail_State data and states shall not be stored in the ONNS source node or any other node. RetrieveDetail_State only operates on stable ONNS connections, with no other ONNS operations pending (including pending RetrieveDetail_Route or RetrieveDetail_State operations). The RetrieveDetail_State request is handled in the same way as the RetrieveDetail_Route request. The only difference is, that the message data from the path nodes and the output to the request contain connection signal states additionally to the connection data as retrieved via RetrieveDetail_Route request.

REPT PTHCHG is not generated

The **`RTRV-PATH`** command does not generate a **`REPT PTHCHG`** message.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No.

Input format

A user defined connection alias (an alternate name) may be optionally specified with the **RTRV-PATH** command. The alias is specified in the spec block as ConnAlias="…" (up to 83 printable ASCII characters + NULL; excluded are the following characters: colon, semicolon, double-quotes, comma). If no *connAlias* is specified, the *connID* is used as the key. If both *connAlias* and *connID* are specified then *connID* is used as the key. If neither is specified, the command is denied.

Command syntax: **RTRV-PATH**:*srcTid:[srcAid]:ctag::[servID]:[connID][:specBlock]*;

Input parameters

Command parameters

1. **srcTid**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **srcAid**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
3. **ctag** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
4. **srvID** – Service Identifier
   This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
   Type: Integer.
   Range: 1 – 20000000
   Only one *srvID* is accepted. The value "ALL" is not supported.

5. **connID** – Connection ID for an existing ONNS optical path connection service This parameter reports only the connections originating at the specified source node. *connID* is formed from the *SrcNodeIP* and IngressPort# and an index # where 1675 LambdaUnite MSS uses bay = 1 = shelf.
   Type: Character string of length up to 39.
   Range: *ALL* – special value to retrieve all connection records
   Range: *nodeID&Tributary AID&Index* - to retrieve a specific entry, with
   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:
     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-1-13-15-1&271584
   Range: *nodeID&Tributary AID range* - if port # specifies a VCG to retrieve a connection group, with:
   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:
     Bay/Frame #(1), Shelf #(1), Slot/Pack #(1..39), Port #(1..9), Tributary #ALL – special value to retrieve all connection records associated with the port ID.

   As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG. The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.
   Range: *nodeID Tributary AID&Index* - to retrieve a protecting connection of an M:N connection group, with:
   - nodeID: four dot separated numeric strings: each string 0..255
   - defaultTributary AID: five dash separated numeric strings with:
     Bay/Frame # (1), Shelf # (1), Slot/Pack # (#), Port # (#), Tributary # (#)
   - Index: Numeric string 0-4294967295
     The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x

   Range: *nodeID&groupindex* -to retrieve all connections belonging to an M:N protected connection group, with:
   - nodeID: four dot separated numeric strings: each string 0..255;
   - groupIndex: integer value specifying a connection group within *nodeID* 0-65535.

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x

Spec block parameters

1. **`DetailRoute`**
   DetailRoute = ON will return the portID / trib information from all nodes in each path of the *connID* specified in the second block of the response message.
   The command is rejected if *DetailRoute* = ON and *connID* = ALL.
   The command is rejected if *DetailRoute* = ON and *connID* contains a tributary ID with qualifier ALL instead of a single tributary number.
   The command is rejected if *DetailState* = ON and *connID* specifies a connection group instead of a single connection.
   Type: Alphanumeric value set.
   Range: OFF, ON.
   Default: OFF.
2. **`DetailState`**
   DetailState = ON will return the portID / trib information including corresponding incoming signal states from all nodes in each path of the connID specified in the second block of the response message.
   The command is rejected if DetailState = ON and connID = ALL.
   The command is rejected, if DetailState = ON and connID contains a tributary ID with qualifier ALL instead of a single Tributary number.
   The command is rejected if DetailState = ON and connID specifies a connection group ID instead of a single connection.
   The command is accepted, if both DetailRoute = ON and DetailState = ON, as far as all other validations are fulfilled. In this case, the action requested by DetailState = ON shall be performed.
   Type: Alphanumeric value set.
   Range: OFF, ON.
   Default: OFF.
3. **`connAlias`** – connection alias
   This parameter specifies a user defined string as an alternate name for the connection instead of *connID*.
   Type: Character string up to length 83.
   Range: The string consists of printable ASCII characters + NULL. The following characters cannot be used in the string: colon, semicolon, comma, double quotes, equal, ampersand, greater than. Additionally the string shall start with a letter.

Output format

Normal system response to RTRV-PATH for connID, output syntax

If the RTRV-PATH request completes successfully, then the following normal completion response is returned to the command.

Note: The *nodeListUsage1* = variable will be translated to either Exclude= or Explicit= , depending on which type of *nodeList* was received as input when the connection was set up.

```
   srcTid date time
```

```
M  ctag COMPLD
   ″ConnID=connID,ConnStatus=connStatus,connControl,pdir,protectionClass,
   bandwidth,[ConnAlias=connAlias],
   MaxDelay=maxDelay,[SrvID=srvID],[nodeListUsage1=nodeList1],
   [nodeListUsage2=nodeList2],[SrlgExclude=srlgExcludeList],
   [IncludeID=linkIncludeID],[specBlock],[reserved],
   SrcNID=srcNID,srcPortID,DestNID=destNID,destPortID
   [,brDestNID=brDestNID,brDestPortID]
   [:PathID=pathID1,State=pathState1,NodeList=pNodeList1
   [:PathID=pathID2,State=pathState2,NodeList=pNodeList2]]″
;
```

NodeList1 = Node45ID & Node12ID & Node39ID & … & Node99ID to be used as set by nodeListUsage1.

pNodeList1 = srcNodeID & Node2ID & Node3ID & … & destNodeID

srlgExcludeList = SRG1 & SRG2 & SRG3 & … & SRG8

brDestNID = brDestNID, brDestPortID to be used for Y-connection.

System response to RTRV-PATH with connID = ALL for an empty connection Data Repository.

ONNS shall respond with a TL1 completion response of COMPLD and a message that the NCDR is empty when there are no connection records in the ONNS Network (Domain) Connection Data Repository (NCDR). If the **RTRV-PATH** ALL request completes successfully with an empty NCDR, then the following completion response is returned to the command.

```
   srcTid date time
M  ctag COMPLD
;
```

System response behavior to RTRV-PATH for connections within an M:N protection group

- For a working connection of an M:N protection group one single path will be associated, when this connection is currently not protected
- For a working connection of an M:N protection group two paths will be associated, when this connection is currently being protected
- For a protecting connection of an M:N protection group one single path will be associated, when this connection is currently not protecting a working connection
- For a protecting connection of an M:N protection group no path at all will be associated, when this connection is currently protecting a working connection

System response to RTRV-PATH with connID [connAlias=connAlias] and DetailRoute=ON

ONNS shall respond with a normal TL1 RTRV-PATH completion response of RTRV and an additional message block with the RetrieveDetail_Route information included. If the RTRV-PATH <connID> request completes successfully with a normal NCDR record as above, then the following completion response is returned to the command:

```
   srcTid date time
M  ctag COMPLD
   ″ConnID=connID,PathID=pathID1[,ConnAlias=connAlias]:
   nodeID1-1,connPtID1_1-1,[protConnPtId_1-1],connInd,connPtID2_1-1,,
[detailRouteStatus]:
   nodeID1-2,connPtID1_1-2,,connInd,connPtID2_1-2,,[detailRouteStatus]:
   nodeID1-3,connPtID1_1-3,,connInd,connPtID2_1-3,,[detailRouteStatus]:
   ...
   nodeID1-n,connPtID1_1-n,,connInd,connPtID2_1-n,[protConnPtID_1-n],
[detailRouteStatus]″
  [″ConnID=connID,PathID=pathID2[,ConnAlias=connAlias]:
    nodeID2-1,connPtID1_2-1,[protConnPtID_2-1],connInd,connPtID2_2-1,,
[detailRouteStatus]:
    nodeID2-2,connPtID1_2-2,,connInd,connPtID2_2-2,,[detailRouteStatus]:
    nodeID2-3,connPtID1_2-3,,connInd,connPtID2_2-3,,[detailRouteStatus]:
    ...
    nodeID2-n,connPtID1_2-n,,connInd,connPtID2_2-n,[protConnPtID_2-n],
[detailRouteStatus]″]
;
```

Note: ConnAlias is reported only for releases in which it is supported as input parameter. See above. It depends on the protectionClass and connStatus whether one or two records are reported: UNPROT connections always have one record, N1PLUS1and YCONN connections always have two records, AUTO connections may have two records (e.g. in case restoration is active for revertive mode) or one record. Protection connections of a M:N protection group which are in use for protection shall response without the RetrieveDetail_Route information.

System response to RTRV-PATH with connID [connAlias=connAlias] and DetailState=ON

ONNS shall respond with a normal TL1 **RTRV-PATH** completion response of RTRV and an additional message block with the RetrieveDetail_State information included. If the **RTRV-PATH <connID>** request completes successfully with a normal NCDR record as above, then the following completion response is returned to the command.

```
   srcTid date time
M  ctag COMPLD
   ″ConnID=connID,PathID=pathID1[,ConnAlias=connAlias]:
   nodeID1-1,connPtID1_1-1,[protConnPtId_1-1],connInd,connPtID2_1-1,,
   connPt1State_1-1,[connPt2State_1-1],[detailRouteStatus]:
```

```
   nodeID1-2,connPtID1_1-2,,connInd,connPtID2_1-2,,
   connPt1State_1-2,[connPt2State_1-2],[detailRouteStatus]:
   nodeID1-3,connPtID1_1-3,,connInd,connPtID2_1-3,,
   connPt1State_1-3,[connPt2State_1-3],[detailRouteStatus]:
   ...
   nodeID1-n,connPtID1_1-n,,connInd,connPtID2_1-n,[protConnPtID_1-n],
   connPt1State_1-n,[connPt2State_1-n],[detailRouteStatus]"
  ["ConnID=connID,PathID=pathID2[,ConnAlias=connAlias]:
   nodeID2-1,connPtID1_2-1,[protConnPtID_2-1],connInd,connPtID2_2-1,,
   connPt1State_2-1,[connPt2State_2-1],[detailRouteStatus]:
   nodeID2-2,connPtID1_2-2,,connInd,connPtID2_2-2,,
   connPt1State_2-2,[connPt2State_2-2],[detailRouteStatus]:
   nodeID2-3,connPtID1_2-3,,connInd,connPtID2_2-3,,
   connPt1State_2-3,[connPt2State_2-3],[detailRouteStatus]:
    ...
   nodeID2-m,connPtID1_2-m,,connInd,connPtID2_2-m,[protConnPtID_2-m],
   connPt1State_2-m,[connPt2State_2-m],[detailRouteStatus]"]
;
```

Normal system response to RTRV-PATH with connID = ALL or with tributary# = ALL

Connection record definition:

ConnectionIDx_dataRecord=
"<Connection_info_data>:<Path_Data_record1>[:<Path_Data_record2>]"

Connection_info_data= connID,connStatus,connControl,pdir,protectionClass,bandwidth,[connAlias],

<ConnConstraint_info>,[specBlock],{reserved},<SourceDest_info>

ConnConstraint_info= maxDelay,[srvID],[nodeListUsage1,nodeList1],[nodeListUsage2,nodeList2],[srlgExcludeList],[linkIncludeID],

SourceDest_info = srcNID,srcPortID,destNID,destPortID[,brDestNID,brDestPortID]

Path_Data_record1= pathID1,pathState1,pNodeList1:

Path_Data_record2= pathID2,pathState2,pNodeList2

If the RTRV-PATH request with connID = ALL or with tributary# = ALL completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   "ConnectionID1_dataRecord"
   ...
   "ConnectionID4_dataRecord"
>
   srcTid date time
M  ctag RTRV
```

```
   ″ConnectionID5_dataRecord″
>
   srcTid date time
M  ctag COMPLD
   ″ConnectionID6_dataRecord″
   ...
   ″ConnectionID11_dataRecord″
;
```

The RTRV-PATH <connID=ALL> command shall send a maximum of 8 connection records per block. On all large output message blocks but the last, the continuation RTRV completion code is used and the semicolon(;) is replaced with a (>) terminator. Then on the last record block, the COMPLD completion code is used with the final semicolon (;) data block terminator.

Note: This RTRV-PATH <connID=ALL> command can produce large amounts of data. This will require proper sizing of the receive buffer of the OS system. brDestNID = brDestNID,brDestPortID to be used for Y-connection.

Wrt. output format for the RTRV-PATH <tributary # = ALL> command the same requirements are valid as for the RTRV-PATH <connID=ALL> command.

Normal system response to RTRV-PATH with connID = M:N groupID

Connection record definition:

ConnectionIDx_dataRecord =
″<Connection_info_data>:<Path_Data_record1>[:<Path_Data_record2>]″

Connection_info_data = connID,connStatus,connControl,pdir,protectionClass,bandwidth, [connAlias], <ConnConstraint_info>,[specBlock],{reserved}, lt;SourceDest_info>

ConnConstraint_info = maxDelay,[srvID],[nodeListUsage1,nodeList1], [nodeListUsage2,nodeList2],[srlgExcludeList],[linkIncludeID],

SourceDest_info = srcNID,srcPortID,destNID,destPortID[,brDestNID,brDestPortID]
Path_Data_record1 = pathID1,pathState1,pNodeList1:

Path_Data_record2 = pathID2,pathState2,pNodeList2

If the RTRV-PATH request with connID = M:N groupID completes successfully, then the following normal completion response is returned:

```
   srcTid date time
M  ctag RTRV
   ″Route1,NumberOfConnections,GroupAsymmetry″
   ″ConnectionID1_dataRecord″
   ...
   ″ConnectionID4_dataRecord″
>
   srcTid date time
```

```
M  ctag RTRV
   ″Route2,NumberOfConnections,GroupAsymmetry″
   ″ConnectionID5_dataRecord″
>
   srcTid date time
M  ctag COMPLD
   ″ConnectionID6_dataRecord″
   ...
   ″ConnectionID8_dataRecord″

   ″Route3,NumberOfConnections,GroupAsymmetry″
   ″ConnectionID9_dataRecord″
   ...
   ″ConnectionID12_dataRecord″
;
```

The RTRV-PATH <connID=M:N groupID> command shall send a maximum of 8 connection records per TL1 block. On all large output message blocks but the last, the continuation RTRV completion code is used and the semicolon(;) is replaced with a (>) terminator. Then on the last record block, the COMPLD completion code is used with the final semicolon (;) data block terminator.

Note: This RTRV-PATH <connID=M:N groupID> command can produce large amounts of data. This will require proper sizing of the receive buffer of the OS system.

Output parameters

Command parameters

1. **`srcTid`** – Source Node Identifier
   See definition of this parameter in Input Parameters section.
2. **`date`**
   This parameter is set to the date when the output message is generate: YY-MM-DD where YY=00-99, MM=01-12, DD=01-31.
3. **`time`**
   This parameter is set to the time when the output message is generated: HH:MM:SS where HH=00-23,MM=00-59, SS=00-59.
4. **`ctag`** – Correlation Tag
   See the parameter description in the Input parameters section above.
5. **`completionCode`**
   Standard TL1 completion code result.
   Type: Value set.
   Range: COMPLD (default) for success: RTRV for data continuation.

6. **`route`** – M:N Route
   Indicates the M:N route in front of a group of M:N protected connections with the same node list. This parameter is only used for retrival of all connections belonging to an M:N protected group.
   Type: Alphanumeric string
   Range: WorkerRoute1, WorkerRoute2, WorkerRoute3, ProtectionRoute1
7. **`numberOfConnections`** – Number of Connections per Route
   NumberOfConnections = x indicates the number of connections per route of a M:N protection group. This parameter is only used for retrival of all connections belonging to an M:N protected group.
   Type: Integer.
   Range: 1-96 (number of connections per route).
8. **`groupAsymmetry`** – M:N Group Asymmetry
   GroupAsymmetry = x indicates the connection asymmetry between the routes of a M:N protection group. This parameter is only used for retrival of all connections belonging to an M:N protected group.
   Type: Integer.
   Range: na = not applicable (value of asymmetry for the protection route); 0 – 255 indicates the difference between the number of actual worker connections within this route and the ceiling average value of worker connections per route based on the protection group.
9. **`connID`**
   See the parameter description in the Input parameters section above.
10. **`connAlias`**
    See the parameter description in the Input parameters section above.
11. **`connStatus`**
    ConnStatus = pst&sst[&tst]
    Connection status is the overall status of the connection *connID* which may be different than the status of an individual *pathID* in that connection.
    connStatus is composed out of
    - pst (PrimaryState)
    - sst (SecondaryState)
    - tst (TertiaryState, optional).

    Up to two of the allowed values are simultaneously permitted as a sst/tst parameter. The two values must be separated by an ampersand (“&”). Individual status values are 7 characters or less.
    Type: up to three &-separated value sets.
    Range *pst* values:
    - CON: Connected
    - DLT: Deleted
    - NOCON: Not Connected

- MAINT: Maintenence Mode
- UNKNOWN: Conn state unknown.

Range *sst* & *tst* values:

- OK: Conn good
- FAILED: Conn failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- PREMPT: Path pre-empted
- SETUPFL: Connection SetUp Failed
- FL-NVM: Not saved in NVM
- WTORV: Waiting to revert path
- EPRTFLEDGE: Port Failed
- BAD-IN Conn failed, Bad signal in
- BRDGD Path bridged

Note: Some states listed in this list will only occur within a DLT-PATH command response or within a REPT PTHCHG or REPT PTHEVT notification message.

Range *sst* values:

- STBY: Standby (for protecting connections within an M:N protection group, when connection is not protecting a working connection)
- USED: Used (for protecting connections within an M:N protection group, when connection is currently protecting a working connection)

Range *tst* values:

- DEGRADED: Non-revertive restoration was not possible if the worker or a permanent 1+1 protected connection fails.

12. **`connControl`** -Connection Control
    This parameter indicates whether the ONNS connection is controlled by ONNS or by external control (UNI, eNNI connections).
    Range:
    - ONNS (ONNS connections, created via ENT-PATH)
    - EXT (ONNS connections, created via ENT-CONN, UNI or eNNI)

13. **`pdir`** – Path Direction Topology
    Unidirectional A-Z (1WAY) or Bidirectional (2WAY).
    For connections of *protectionClass* MBYNWRK and MBYNPRT only the path direction topology Bidirectional (2WAY) is supported.
    Type: Value set.
    Range: 1WAY, 2WAY
    Default: 1WAY.
14. **`protectionClass`**
    This parameter defines the type of restoration protection assigned to the path connection during path setup or after path modification.

    Range:

    - UNPROT (path from source to destination is un-protected)
    - AUTONR (path from source to destination is a non-revertive auto-reroute path connection)
    - N1PLUS1 (paths from source to destination is set up as a 1+1 protected connection)
    - AUTORV (path from source to destination is a revertive auto-reroute path connection)
    - YCONN (paths from the same source to two different destinations are set up as a bridged connection).
    - MBYNWRK (working connection in an M:N connection group)
    - MBYNPRT (protecting connection in an M:N connection group)
    - P1PLUS1NR (paths from source to destination are set up as a 1+1 protected non-revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)
    - P1PLUS1NRRV (paths from source to destination are set up as a 1+1 protected revertive connection with an additional non-revertive restoration path applied in case the worker or protection path fails.)

    Default: None.
15. **`bandwidth`** – Requested bandwidth / signalRate
    Link *signalRate* is equal to or greater than the minimum cross-connect rate *minXCrate* of the *port_type*.
    Type: Value set.
    Range: STS1, STS3, STS12 STS48, STS192, STS1-pipe, STS3-pipe, STS12-pipe STS48-pipe, STS192-pipe.
    Default: None.
16. **`maxDelay`** – Maximum transport delay
    Maximum transport delay through all nodes in the path. Total path delay must not exceed this value.
    Range: 0, is used to select no delay limit; 1 – 32767, maximum transport delay in milliseconds.
    Default: 0.

17. **`srvID`** – Service Identifier
    This parameter is a logical tag for the group of connections associated with a service defined by the Service Manager system (or other management entity).
    The *srvID* parameter is an integer value range = 1 – 20,000,000. Only one *srvID* is accepted. ALL is not supported.

18. **`nodeListUsage1`**
    For connection setup NN used/uses the list of nodes included in the *nodeList1* array block in a manner specified by the *nodeListUsage1* variable. Both, *nodeListUsage1* and *nodeList1*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command. An EXCLUDE list usage applies to all paths created (as in a N1PLUS1NR or N1PLUS1RV). A non-existent *nodeID* will be ignored in EXCLUDE *nodeList1* during path setup. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
    Type: Value set.
    Range:
    - EXPLICIT, *nodeList1* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.)
    - EXCLUDE, nodes in the *nodeList1* must all be excluded from the path created. In the case when the *protectionClass* is either YCONN, N1PLUS1NR or N1PLUS1RV the nodes in the *nodeList1* below also apply to the second or restoration path.

19. **`nodeList1`**
    When *nodeListUsage1* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([&nid2& …&nidx]).
    When *nodeListUsage1* = EXCLUDE: List of nodes that must all be excluded from both paths. The *nodeID* values are shown in the from: nid1([&nid2& …&nidx]).
    If the preceding required *nodeListUsage1* field is not supplied, this field must not be present. If user specifies empty EXCLUDE *nodeList1*, the connection should use indirect path from source node to destination node (that is, at least one intermediate node is needed). If user specifies empty EXPLICIT *nodeList1*, the connection should use a direct path from source to destination node (that is, no intermediate node is allowed).
    Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
    Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
    Range: nid1([&nidx])* or empty list (where X <= 25 when *nodeListUsage1* = EXPLICIT) (where X <= 25 when *nodeListUsage1* = EXCLUSIVE)

20. **`nodeListUsage2`**
    For connection setup NN used/uses the list of nodes included in the *nodeList2* array block in a manner specified by the *nodeListUsage2* variable. Both, *nodeListUsage2* and *nodeList2*, fields belong together and is reported when supplied as input to the **`ENT-PATH`** command. A non-existent *nodeID* will cause setup failure in EXPLICIT *nodeList1* during path setup.
    This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
    This parameter shall also be allowed when *protectionClass* is P1PLUS1NR.
    Type: Value set.
    Range: EXPLICIT, *nodeList2* contains the intermediate NE nodes of the path in the explicit order they are to be used. (An empty list will force a direct path from source to destination.).
21. **`nodeList2`**
    When *nodeListUsage2* = EXPLICIT: List of intermediate nodes that specify the path in the exact node order specified in the input. Source and destination nodes are assumed to be included implicitly. The *nodeID* values are shown in the form: nid1([[&nid2& …&nidx]]).
    If the preceding required *nodeListUsage2* field is not supplied, this field must not be present. If user specifies empty EXPLICIT *nodeList2*, the connection should use a direct path from source to destination node (i.e., no intermediate node is allowed).
    This parameter is only be used when *protectionClass* is N1PLUS1 or YCONN.
    This parameter shall also be allowed when *protectionClass* is P1PLUS1NR
    Type: &-separated list of *nodeIDs*. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
    Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123
    Range: nid1([&nidX])* or empty list where X <= 8 when *nodeListUsage1* = EXPLICIT.
22. **`srlgExcludeList`** – SRLG Exclude List
    List of Shared Risk Link Group identifiers that are to be used to exclude links from the connection path(s) being created.
    Type: &-separated list of numeric strings | empty | 0
    Example: 33&49&64
    Range: up to 25 numeric strings, each in the range 1 – 65535. The value 0 (default) or an empty field indicates no constraint are applied.
23. **`linkIncludeID`** – VPN Identifier
    A connection constraint link property such as *linkIncludeID* that must be valid to include the link in the connection path(s) to be created. *linkIncludeID* is a link connection property that can be used to group and classify links. VPNID is a special case of this associative property.
    Type: Integer
    Range: 1 … 65535; 0 or an empty field indicates no VPN constraint is applied.
    Note: Since the corresponding value is not supported as Host port parameter any input different from 0 will lead to DENY of the command.

24. **srcNID** – Source nodeID
ONNS source nodeID. Together with the following srcPortID field defines the associated Ingress connPoint that connects the source node to the edge/gateway switch or client router.
Range: Four dot separated numeric strings. 0 – 255 for each numeric string.
25. **srcPortID** – Source PortID
ONNS source node ingress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the *srcNID* ingress *connPoint*. ONNS assumes a bidirectional port pair.
Value: dash-separated list of 5 numeric strings
Example: 1-1-3-1-4
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port # #, Tributary ##. This range is valid for protecting connections in M:N connection groups
26. **destNID** – Destination nodeID
ONNS destination nodeID. Together with the following *destPortID* field defines the associated Engress *connPoint* that connects the destination node to the edge/gateway switch or client router.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.
27. **destPortID** – Destination PortID
ONNS destination node egress portID. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the destNID egress *connPoint*.
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack ##, Port ##, Tributary ##. This range is valid for protecting connections in M:N connection groups
See the parameter description in the "ENT-PATH: Create ONNS path connection" (p. 3-1701) Input parameters section.
28. **brDestNID** – Bridged Destination NodeID
ONNS bridged destination nodeID. Together with the following brDestPortID field defines the second associated Egress connPoint that connects the destination node to the edge/gateway switch or client router in case of protectionClass YCONN.
Type: Four dot separated numeric strings. 0 – 255 for each numeric string.
29. **brDestPortID** – Bridged Destination PortID
ONNS bridged destination node egress *portID* in case *protectionClass* = YCONN. The *portID* consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # of the brDestNID egress connPoint. 4.1; 5; 6
Value: dash-separated list of 5 numeric strings
Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).

30. **`pathID1`** – Path ID 1
    The PathID of the 1st path that was created.
    Type: character string of length up to 20.
    Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255) (Index: Numeric string 0-65535).
    The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x
    Example: 29.13.196.203-65111
    Default: None.

31. **`pathState1`** – Path State 1
    Path State1 = *pst*& *sst*[*&tst*]. pathState1 represents the status of the first path of the connection which may be different than the status of the overall connection in *connStatus*.
    PathState1 is composed out of *pst* (PrimaryState), *sst* (SecondaryState) and *tst* (TertiaryState, optional) Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
    Type: up to three &-separated value sets.

    Range *pst* values:

    - CON: Connected
    - NOCON: Not Connected
    - DLT: Deleted
    - MAINT: Maintenance mode.

    Range *sst* & *tst* values

    - OK: Conn good
    - FAILED-PATH: failed
    - SETUP: Conn SetUp occurring
    - TRDOWN: Conn teardown occurring
    - NO-IN: Conn failed, signal in bad
    - RSTOR: path restore occurring
    - RSTRD: Conn restored
    - RSTRFL: Conn restore failed
    - DLT-FL: Delete operation Failed
    - UNKNOWN: Conn state unknown
    - WTORSTR: Waiting to restore path
    - SETUPFL: Connection SetUp Failed
    - RVRTFL: Conn revert failed
    - PREMPTD: Preempted
    - WTORV: Waiting to revert path
    - BRDGD: Path bridged

- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

Range *pst* values:

- RSVD: Reserved, for paths associated to protecting connections within an M:N protection group, while not protecting a working traffic.

32. **`pNodeList1`** – Path Node List 1

The path node list contains the list of nodes that the 1st path (*pathID1*) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A nodeID is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

33. **`pathID2`** – Path ID 2

The PathID of the 2nd path (if existing) that was created.
Type: character string of length up to 20.
Range: nodeID-Index (nodeID: four dot separated numeric strings: each string 0 – 255), (Index: Numeric string 0-65535).
The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x.
Example: 29.13.196.203-65111

34. **`pathState2`** – Path State 2

pathState2 = pst&sst[&tst]. *pathState2* represents the status of the second path (if existing) of the connection which may be different than the status of the overall connection in *connStatus*.

PathState2 is composed out of

- pst (PrimaryState)
- sst (SecondaryState)
- tst (TertiaryState, optional).

Up to two of the allowed values are simultaneously permitted as a sst/tst parameter.The two values must be separated by an ampersand (″&″). Individual status values are 7 characters or less.
Type: up to three &-separated value sets

Range *pst* values:

- CON: Connected
- NOCON: Not Connected
- DLT: Deleted
- MAINT: Maintenance mode.

Range *sst* & *tst* values

- OK: Conn good
- FAILED-PATH: failed
- SETUP: Conn SetUp occurring
- TRDOWN: Conn teardown occurring
- NO-IN: Conn failed, signal in bad
- RSTOR: path restore occurring
- RSTRD: Conn restored
- RSTRFL: Conn restore failed
- DLT-FL: Delete operation Failed
- UNKNOWN: Conn state unknown
- WTORSTR: Waiting to restore path
- SETUPFL: Connection SetUp Failed
- RVRTFL: Conn revert failed
- PREMPTD: Preempted
- WTORV: Waiting to revert path
- BRDGD: Path bridged
- EPRTFL: EDGE Port Failed
- BAD-IN: path failed, Bad signal in
- FLT: Fault.

Note: Some states listed in this list will only occur within a DLT-PATH command response or in a REPT-PTHCHG / PTHEVT notification.

35. **`pNodeList2`** – Path Node List 2
The path node list contains the list of nodes that the 2nd path (pathID1, if existing) traverses in A-Z order from the source node.
Type: &-separated list of nodeIDs. A *nodeID* is represented as four dot separated numeric string with each numeric string between 0 – 255.
Range: nid1([&nidX])*
Example: 135.13.196.203&135.13.42.23&135.213.242.133&135.13.196.123

Spec block parameters

1. **`connMode`** – Connection Mode
The connection mode determines ONNS treatment of the failed path after an auto-reroute restoration.
Example: *connMode* = NONREVERTIVE causes path setup to be auto-reroute non-revertive.
Range:

- UNPROT: converts a auto connection to a unprotected
- N1PLUS1: Network 1+1 protected connection

- NONREVERTIVE: Auto-reroute non-revertive
- REVERTIVE: Auto-reroute revertive
- YCONN: Y-connection
- BRIDGEY: Y connection Protection
- MBYNWRK: Working connection within an M:N protection group
- MBYNPRT: Protecting connection within an M:N protection group.

2. **`rstrPriority`** – Restoration Priority
   This parameter describes the restoration priority of the connection.
   Range: 0 (default) – 7. Lower values represent higher priority.
3. **`DetailRoute`**
   DetailRoute = ON will return the portID / trib information from all nodes in each path of the *connID* specified in the second block of the response message.
   The command is rejected if *DetailRoute* = ON and *connID* = ALL.
   The command is rejected if *DetailRoute* = ON and *connID* contains a tributary ID with qualifier ALL instead of a single tributary number.
   The command is rejected if *DetailState* = ON and *connID* specifies a connection group instead of a single connection.
   Type: Alphanumeric value set.
   Range: OFF, ON.
   Default: OFF.
4. **`DetailState`**
   DetailState = ON will return the portID / trib information including corresponding incoming signal states from all nodes in each path of the connID specified in the second block of the response message.
   The command is rejected if DetailState = ON and connID = ALL.
   The command is rejected, if DetailState = ON and connID contains a tributary ID with qualifier ALL instead of a single Tributary number.
   The command is rejected if DetailState = ON and connID specifies a connection group ID instead of a single connection.
   The command is accepted, if both DetailRoute = ON and DetailState = ON, as far as all other validations are fulfilled. In this case, the action requested by DetailState = ON shall be performed.
   Type: Alphanumeric value set.
   Range: OFF, ON.
   Default: OFF.
5. **`connectionPriority`** – Connection Priority
   Connection priority allows connection setup of a higher priority connection to utilize the existing path through the network when the existing connection is pre-emptible.
   Range: 0 (default) – 7. 0 = locked, 1 – 7 = Priority, Connection may be pre-empted.

6. **`disjointness`**
   The actual level of disjointness of the calculated and established connection. The parameter is not filled when protectionClass is UNPROT.
   Type: Alphanumeric value set.
   Range:
   - COMPLETE: fully node and SRLG disjoint path
   - FULLYSRLG: maximal node and fully SRLG disjoint path
   - MAXIMAL: maximal node disjoint path.
7. **`connGrpID`**
   Connection Group ID for an existing ONNS connection to report the connection group the connection is member of or ″UNDEFINED″, if connection is not member of a group.
   For connection roups identified by A VCG EDGE port, connGrpID is formed from the *SrcNodeID* and the Ingress Port #, where 1675 LambdaUnite MSS uses bay =1= shelf.
   The connection group ID is automatically assigned to a connection, when it is setup. The connection group ID will consist of the *nodeID* and the EDGE port AID in case the EDGE port is a VCG. The connection group ID will be ″UNDEFINED″ otherwise. For M:N protected connection groups, *connGrpID* is formed from the *SrcNodeID* and a group index chosen by the source node to be unique within this node. The connection group ID is automatically assigned to a connection, when it is setup. The connection group ID will consist of the node ID and a group index in case the connection is member of an M:N connection group. The connection group ID will be ″UNDEFINED″ otherwise.
   Type: Character string of length up to 25.
   Range: UNDEFINED - special value for connections not being member of a group nodeID&Prt AID - value for connections being member of a ″GbE Connection Group″ nodeID&groupIndex - value for connections being member of a ″M:N Connection Group″
   - nodeID: four dot separated numeric strings: each string 0..255
   - Port AID: four dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port # with: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1..39), Port # (1..8).
   - groupIndex: integer value specifying an M:N connection group 0-65535.
     The nodeID for 1675 LambdaUnite MSS starts always with 29.x.x.x

8. **protRatio** – Protection ratio
   For each working or protecting connection of an M:N protected connection group this parameter will report the concrete protection ratio as specified on creation of the group. For connections not being member of an M:N protected connection group, this parameter is not reported at all.
   Type: Alphanumeric value set
   Range. BY1 (ratio of 1 to 1 in number of protecting and working connections), 1BY2 (ratio of 1 to 2 in number of protecting and working connections), 1BY3 (ratio of 1 to 3 in number of protecting and working connections).
9. **assocConnID** – Associated Connection ID
   Associated Connection ID for an existing ONNS connection of an M:N connection group to report the connection ID of the associated connection within the group, when a protection switch is active, i.e. for a working connection the ID of the associated protecting connection and for a protecting connection the ID of the associated working connection is reported. The associated connection ID will be ″OPEN″ for connections of an M:N connection group not involved in a protection switch. The associated *connectionID* will not be reported for connections not being member of an M:N connection group.
   Type: Character string of length up to 39.

   Range: ALL - special value to retrieve all connection records | nodeID&Tributary AID&Index - to retrieve a specific entry, with

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

     Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 192).
   - Index: Numeric string 0-4294967295

   The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x.
   Example: 29.13.196.203&1-1-13-15-1&271584

   nodeID&Tributary AID range - if port # specifies a VCG to retrieve a connection group, with:

   - nodeID: four dot separated numeric strings: each string 0..255
   - Tributary AID: five dash separated numeric strings representing Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary # with:

     Bay/Frame #(1), Shelf #(1), Slot/Pack #(1..39), Port #(1..9), Tributary #ALL – special value to retrieve all connection records associated with the *portID*.

   As there is always at most one connection group associated with a VCG, the qualifier ALL specified for the Tributary # of input parameter *connID* will always deliver all connections of the affected group, i.e. all connections starting at the affected VCG. The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x Example: 29.13.196.203&1-5-13-7-ALL. The port AID 1-5-13-7 must denote a VCG.

nodeID Tributary AID&Index - to retrieve a protecting connection of an M:N connection group, with:

- nodeID: four dot separated numeric strings: each string 0..255
- defaultTributary AID: five dash separated numeric strings with:
  Bay/Frame #(1), Shelf #(1), Slot/Pack ##, Port ##, Tributary ##
- Index: Numeric string 0-4294967295

  The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x

  nodeID&groupindex -to retrieve all connections belonging to an M:N protected connection group, with:

  nodeID: four dot separated numeric strings: each string 0..255; groupIndex: integer value specifying a connection group within *nodeID* 0-65535.

  The *nodeID* for 1675 LambdaUnite MSS starts always with 29.x.x.x

10. **`htmr`** – Hold-Off Time

    This parameter specifies the value of the hold-off timer for 1+1 protection switching.
    Parameter applies to a permanent 1+1 protected connection.
    Parameter does not apply to a 1+1 protected connection.
    Parameter applies to 1+1 protected and permanent 1+1 protected connections.
    Type: numeric string [ms].
    Range for high order path protection groups: 0 .. 99 ms in 1 ms steps and 100 .. 10000 ms in 100 ms steps.
    Range for low order path protection groups: 0, 25, 50, 75 ms (in steps of 25 ms) and 100 .. 10000 ms in 100 ms steps.
    Default:
    None.
    The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block. Default Value: 0
    This parameter shall relate to the *htmr* parameter provisioned via the TL1 commands **`ENT-PATH`** or **`ED-PATH-PROT`**. Configuration of path protection group parameter via **`ED-PROTN-GRP`** is not forbidden, but should be discouraged, since this configuration is not reflected in the ONNS view. During normal operation of an ONNS connection, the **`ED-PROTN-GRP`** parameters are considered but during ONNS protection conversion, the ONNS stored path protection group parameters are considered.

11. **`wtr`** – Wait To Restore Time

    This parameter specifies the wait-to-restore time for revertive 1+1 protection switching.
    Parameter does not apply to a 1+1 protected connection revertive.
    Parameter applies to a permanent 1+1 protected connection with revertive SNCP/UPSR protection.
    Type: numeric string [minute].
    Range: 0 – 60.
    Default:
    None.

The path protection groups at the A and Z nodes of the connection are created with default values, if the parameter is not present in the spec_block. Default Value: 0 This parameter shall relate to the *wtr* parameter provisioned via the TL1 commands **ENT-PATH** or **ED-PATH-PROT**. Configuration of path protection group parameter via **ED-PROTN-GRP** is not forbidden, but should be discouraged, since this configuration is not reflected in the ONNS view. During normal operation of an ONNS connection, the **ED-PROTN-GRP** parameters are considered but during ONNS protection conversion, the ONNS stored path protection group parameters are considered.

12. **switchBack** – revertive restoration path switch back
    This parameter describes how a revertive restoration path is switched back to the original path.
    Type: Alphanumeric value set.
    Range:
    - AUTO: path is automatically switched back after wait to revert time.
    - MAN: path has to be manually switched back triggered by the TL1 command **ED-PATH-ROLL**.

    Default: AUTO.
13. **detailPath Record**
    One or two detailPath records will be reported when *detailRoute* = ON.
    a. **connID** – Connection ID
       See description in the Input parameters section below.
    b. **pathID** – Path ID
       See description in the section “Output parameters” below.
    c. **connAlias** – connection alias
       See description in the Input parameters section below.
    d. **xcon Record**
       An xcon record will be reported for each node in the path when *DetailRoute* = ON. Records are ordered in A-Z direction from the source node:
       nodeID: ID of the node which reports the xcon record. Type: Four dot separated numeric strings. Range: 0 – 255 for each numeric string.
       connPtID1 – Connection Point ID 1: ONNS connection point identifier for the ingress connection point of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #. Type: dash-separated list of 5 numeric strings. Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).
       Slot/Pack ##, Port ##, Tributary ##. Value # for Slot/Pack, Port and Tributary indicates default tributary AIDs used for reporting open ends in not existing xcons in the A-node or Z-node for a protecting NN connection of an M:N protected connection group.
       connInd – Connection Indicator: Indicator whether cross-connection is setup. Type: Alphanumeric value set. Range: -X- (cross-connection setup).

connPtID2 – Connection Point ID 2: ONNS connection point identifier for the egress connection point of the cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #. Type: dash-separated list of 5 numeric strings. Range: Bay/Frame # (1), Shelf # (1), Slot/Pack # (1 – 39), Port # (1 – 16), Tributary # (1 – 128).

protConnPtID – Protection Connection Point ID: ONNS connection point identifier for the ingress or egress connection point of the protected OUC cross-connection. It consists of Bay/Frame #, Shelf #, Slot/Pack #, Port #, Tributary #. Type: dash-separated list of 5 numeric strings. Range: Bay/Frame #(1), Shelf #(1), Slot/Pack #(1..39), Port #(1..16), Tributary #(1..128).

detailRouteStatus – Detail Route Node/XC Status: This parameter indicates the status of a node and/or its cross-connection. Range: NO_REPLY (no reply from node) (no cross-connection found).

connPt1State – Fault State of Incoming Signal at Connection Point 1: This parameter indicates the fault state monitored for the incoming signal at the tributary identified by Connection Point ID 1.

Type:

Alphanumeric value set.

Range:

UNKNOWN (the incoming signal could not be monitored); OK (a valid incoming signal is monitored); FAILED (no valid incoming signal is monitored, but in corr. direction an incoming signal state OK is monitored at the EDGE tributary); NO-IN (no valid incoming signal is monitored and in corr. direction an incoming signal state FAILED is monitored at the EDGE tributary).

Context wrt. OUCs:

In case an OUC is configured on top of the ONNS connection in the A-node, *connPt1State* shall denote the signal state of the EDGE tributary currently selected by the corresponding path selector. That means that in case of an OUC configured on top of an ONNS connection, it is not possible to get the signal state of both involved EDGE tributaries, but the signal is monitored behind the selector, thus yielding the signal state of the tributary currently selected.

connPt2State – Fault State of Incoming Signal at Connection Point 2: This parameter indicates the fault state monitored for the incoming signal at the tributary identified by Connection Point ID 2.

Type:

Alphanumeric value set.

Range:

UNKNOWN (the incoming signal could not be monitored); OK (a valid incoming signal is monitored); FAILED (no valid incoming signal is monitored, but in corr. direction an incoming signal state OK is monitored at the EDGE tributary); NO-IN (no valid incoming signal is monitored and in corr. direction an incoming signal state FAILED is monitored at the EDGE tributary).

Context wrt. OUCs:

In case an OUC is configured on top of the ONNS connection in the A-node, *connPt2State* shall denote the signal state of the EDGE tributary currently selected by the corresponding path selector. That means that in case of an OUC configured on top of an ONNS connection, it is not possible to get the signal state of both involved EDGE tributaries, but the signal is monitored behind the selector, thus yielding the signal state of the tributary currently selected.

Example input/output

Normal system response to RTRV-PATH with connID = ALL with no NCDR records

If the **RTRV-PATH ALL** request completes successfully with an empty NCDR then the following normal completion response is returned to the command:

```
RTRV-PATH:srcTid:srcAid:ctag::srvID:ALL;.

   srcTid date time
M  ctag COMPLD
;
```

Input/Output example for a auto-reroute connection

The following example shows the successful completion of a **RTRV-PATH** command by the network element for a (non-revertive) auto-reroute connection:

```
RTRV-PATH:LT-ONNS-789:snn:193456::1351379:135.13.196.203&1-6-5-3-1&1;

   LT-ONNS-789 2001-01-01 08:00:00
M  193456 COMPLD
   ″ConnID=135.13.196.203&1-6-5-3-1&1,ConnStatus=CON&OK,ONNS,1WAY,AUTONR,STS48,
   ConnAlias=AEXP-NY-BOS-ckt,
   MaxDelay=9,SrvID=1351379,Explicit=135.13.42.3&135.213.242.123&135.13.198.2,,
   SrlgExclude=1&8&26&33&97,,RstrPriority=3&ConnMode=NONREVERTIVE,,
   SrcNID=135.13.196.203,1-6-5-3-1,DestNID=135.13.242.103,1-2-5-5-1:
   PathID=135.13.196.203-432165,State=CON&OK,
   NodeList=135.13.196.203&135.13.196.23&135.13.42.3&135.13.98.22&135.13.242.103″
;
```

Input/output example for a Network 1+1 connection

The following example shows the successful completion of a **RTRV-PATH** command by the network element for an 1+1 connection:

```
RTRV-PATH:LT-ONNS-9:snn:14345::13513794:135.13.196.203&1-6-5-3-1&1;

   LT-ONNS-9 2001-06-01 08:00:00
M  14345 COMPLD
   ″ConnID=135.13.196.203&1-6-5-3-1&1,ConnStatus=CON&OK,ONNS,2WAY,N1PLUS1,STS48,
   ConnAlias=AEXP-NY-BOS-ckt,
```

```
   MaxDelay=55,SrvID=13513794,Explicit=135.13.196.123&135.13.42.3,Explicit=
      135.13.198.23&135.13.196.3&135.13.198.213,SrlgExclude=135&1432&222,,,,
   SrcNID=135.13.196.203,1-6-5-3-1,DestNID=135.13.196.123,1-2-5-5-1&1-2-5-5-1:
   PathID=135.13.196.203-432165,State=CON&OK,
   NodeList=135.13.196.203&135.13.196.123&135.13.42.3&135.13.242.103:
   PathID=135.13.196.203-432177,State=CON&FAILED,
   NodeList=135.13.196.203&135.13.198.23&135.13.196.3&135.13.198.213&135.13.242.123"
;
```

This connection was setup using the command:

```
ENT-PATH-N1PLUS1:LT-ONNS-9:snn:99345::13513794:2WAY:sts48:55:
   135.13.196.203: 1-6-5-3-1:135.13.196.123:1-2-5-5-1:Explicit,
   135.13.196.123&135.13.42.3:Explicit,
   135.13.198.23& 135.13.196.3&
   135.13.198.213:135&1432&222,10;
```

RTRV-PATH output example for a connection with Path DetailRoute=ON

The following example shows the *detailPath* record of a **RTRV-PATH** command with the parameter *DetailRoute* = ON with an OUC existing on top of the connection in the Z-node It produces this resulting second message block that follows the normal **RTRV-PATH** response (see below for a complete example):

```
RTRV-PATH:LT-ONNS-9:onn:14345::13514794:135.14.196.203&1-6-5-3-1&1:DetailRoute=ON;

...

   LT-ONNS-9 2002-06-04 08:00:00
M  14345 COMPLD
   "ConnID=135.14.196.203&1-6-5-3-1&1,PathID=135.14.196.203-32165,:
   135.14.196.203,1-6-5-3-1,,-X-,1-2-4-3-1,,:
   135.13.196.123,1-1-5-3-1,,-X-,1-2-2-3-1,,:
   135.13.42.63,1-1-3-1-1,,-X-,1-1-4-4-1,,:
   135.13.18.23,1-1-4-4-1,,-X-,1-2-1-3-1,,:
   135.13.242.103,1-1-2-7-1,,-X-,1-2-5-1-1,3-1-2-1-1,"
;
```

RTRV-PATH output example for a connection with Path DetailState=ON

The following example shows the detailPath record of a **RTRV-PATH** command with the parameter *DetailState* = ON, it produces this resulting second message block that follows the normal **RTRV-PATH** response (see below for a complete example):

```
RTRV-PATH:LT-ONNS-9:onn:14345::13514794:135.14.196.203&1-6-5-3-1&1:DetailState=ON;
```

```
...

   LT-ONNS-9 2002-06-04 08:00:00
M  14345 COMPLD
   ″ConnID=135.14.196.203&1-6-5-3-1&1,PathID=135.14.196.203-32165,:
   135.14.196.203,1-6-5-3-1,,-X-,1-2-4-3-1,,
   GOOD,FAILED,:
   135.13.196.123,1-1-5-3-1,,-X-,1-2-2-3-1,,
   GOOD,FAILED,:
   135.13.42.63,1-1-3-1-1,,-X-,1-1-4-4-1,,
   GOOD,FAILED,:
   135.13.18.23,1-1-4-4-1,,-X-,1-2-1-3-1,,
   GOOD,GOOD,:
   135.13.242.103,1-1-2-7-1,,-X-,1-2-5-1-1,,
   GOOD,GOOD,″
;
```

This example output allows the following conclusions:

- In downstream direction, a good signal is received at the A-node EDGE tributary and stays good through the domain
- In upstream direction, a good signal is received at the Z-node EDGE tributary and stays good up to node 135.13.18.23, but is reported as FAILED by node 135.13.42.63. Thus in between these nodes a transmission failure can be assumed.

Output Example for ONNS nodes not reporting or reporting missing cross-connections

This example shows the *detailPath* record of an **RTRV-PATH** with *DetailRoute* = ON, when intermediate nodes do not respond or can not retrieve the cross connect information (see below for a complete example):

```
RTRV-PATH:LT-ONNS-9:onn:14345::13514794:135.14.196.203&1-6-5-3-1&1:DetailRoute=ON;

...


   LT-ONNS-9 2002-06-04 08:00:00
M  13456 COMPLD
   ″ConnID=135.14.196.203&1-6-5-3-1&1,PathID=135.14.196.203-32165,:
   135.14.196.203,1-6-5-3-1,-X-,1-2-4-3-1,:
   135.13.22.3, ,-X-, ,NO_REPLY:
   135.13.196.123,1-1-5-3-1,-X-,1-2-2-3-1,:
   135.13.44.93,1-2-2-2-1,-X-,1-1-4-2-1,:
   135.13.42.63,1-1-4-4-1,-X-,1-2-1-3-1,:
   135.13.242.103,1-1-2-7-1,-X-,1-2-5-1-1,″
;
```

RTRV-PATH output example for an M:N protected group of connections

The example shows a successful completion of **RTRV-PATH** command for an M:N protected group of connections within a state where a protection is currently active:

```
RTRV-PATH:LT-NN-789:snn:973456::1352463:135.13.196.203&1
;

   LT-NN-789 2002-09-18 08:00:00
M  973456 COMPLD

   ”WorkerRoute1,NumberOfConnections=1,GroupAsymmetry=0”
   ”ConnID=135.13.196.203&1-1-13-4-1&1,ConnStatus=CON&RSTRD,ONNS,2WAY,MBYNWRK,
   STS1,ConnAlias=AmericanExp-NY-BOS-ckt_wrk_001,
   MaxDelay=65,SrvID=135246,,,,,
   ProtRatio=1BY2&ConnGrpID=135.13.196.203&1&
   AssocConnID=135.13.196.203&1-1-#-#-#&3,,
   SrcNID=135.13.196.203,1-1-13-4-1,DestNID=135.13.196.123,1-1-5-5-5:
   PathID=135.13.196.203-43216,State=CON&FAILED,
   NodeList=135.13.196.203&135.13.42.3&135.213.242.133&135.13.196.223”
   PathID=135.13.196.203-43218,State=CON&OK,
   NodeList=135.13.196.203&222.111.22.2&141.235.1.17&135.13.196.223”

   ”WorkerRoute2,NumberOfConnections=1,GroupAsymmetry=0”
   ”ConnID=135.13.196.203&1-1-13-4-3&2,ConnStatus=CON&OK,ONNS,2WAY,MBYNWRK,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_wrk_002,
   MaxDelay=65,SrvID=135246,,,,,
   ProtRatio=1BY2&ConnGrpID=135.13.196.203&1&AssocConnID=OPEN,,
   SrcNID=135.13.196.203,1-1-13-4-3,DestNID=135.13.196.123,1-1-5-5-6:
   PathID=135.13.196.203-43217,State=CON&OK,
   NodeList=135.13.196.203&142.330.17.1&222.177.23.4&135.13.196.223”

   ”ProtectionRoute1,NumberOfConnections=1,GroupAsymmetry=na”
   ”ConnID=135.13.196.203&1-1-#-#-#&3,ConnStatus=CON&USED,ONNS,2WAY,MBYNPRT,STS1,
   ConnAlias=AmericanExp-NY-BOS-ckt_prt_001,
   MaxDelay=65,SrvID=135246,,,,,
   ProtRatio=1BY2&ConnGrpID=135.13.196.203&1&
   AssocConnID=135.13.196.203&1-1-13-4-1&1,,
   SrcNID=135.13.196.203,1-1-#-#-#,DestNID=135.13.196.123,1-1-#-#-#”
;
```

Error responses

Validation rule: *connID* Non-Existant (ICNE)

If the source node receives a command and the specified *connID* does not exist, then the following error response is returned:

Error code: `E220 – 13513794 – Input, ConnectionID Non-Existent`

Validation rule: Invalid *srvID*

If the source node receives a command with an invalid ServiceID (srvID) parameter value, then the following error response is returned:

Error code:
`E030 – SMcon135*7.92468 – Syntax, Parameter Data Imvalid,Invalid_ parameter=SrvID`

Validation rule: Invalid command

If an unrecognized or unsupported command is received, then the following error response is returned:

Error code: `E235 – Input, Invalid CoMmand, Command not found or recognized`

Validation rule: Syntax, Invalid punctuation

If a command is received that contains incorrect punctuation, then one of the following error responses is returned:

Error code: `E240 – Syntax, InValid Punctuation`

Validation rule: Syntax invalid or unimplemented parameter

If a command is received that contains an empty field that requires a parameter, or if a parameter input has not an allowed value for the field, then the following error response is returned:

Error code: `E246 – Syntax, Parameter Not Implemented`

The following list identifies conditions that will cause similar error responses :

Error code:
`E030 – SPDI Error – Syntax, Parameter Data Invalid, invalid_parameter= fieldName`

Error code:
`E037 – INCS Error – Input, Neither ConnID nor ConnAlias Specified`

Error code: `E040 – SDNC Error – Syntax, Data Not Consistent`

Error code: `E221 – IANE Error – Input, ConnectionAlias Non-Existent`

Validation rule: Invalid target identifier

If a command is received with an invalid Target Identifier, then one of the following error responses is returned:

Error code: `E250 – Input, Invalid TArget Identifier`

Error code: `E805 – Input, Missing Target Identifier (TID)`

Validation rule: Invalid access identifier

If a command is received with an invalid access identifier, then the following error response is returned:

Error code: `E260 – Input, Invalid ACcess identifier`

Validation rule: Invalid correlation tag (CTAG) or RequestID

If a command is received with an invalid correlation tag, then one of the following error responses is returned:

Error code: `E265 – Input, Invalid Correlation Tag`

Error code: `E266 – Input, Missing Correlation Tag`

Validation rule: Status, All Resources Busy

If a command can not be completed due to insufficient processing resources or the system is in maintenance mode, then the following error response is returned:

Error code: `E280 – Status, All Resources Busy`

The following list identifies conditions that will cause similar error responses:

- Error code:
  `E281 – Status, Busy, NVM Write in Progress for same connection`
- Error code: `E282 – Status, Conn State Invalid for this Operation`

Validation rule: Requested Operation Failed (SROF)

If a command can not be completed due an internal hardware or software failure, then the following error response is returned:

Error code: `E290 – Status, Requested Operation Failed`

Example: The command
`RTRV-PATH:LT-ONNS9:snn:123456:::123.198.42.233&1-2-32-4-1;` can produce this output.

Validation rule: Error Responses Specific to RTRV-PATH

The following list identifies conditions that will cause error responses specific to **`RTRV-PATH`** commands:

Error code:
`E226 – IDNA – Input, Retrieve Detail_Route or Detail_State Not allowed for RTRV-PATH ALL`

The following list identifies conditions that will cause error responses specific to **`RTRV-PATH`** commands sent with detail_route option:

Error code:
`E227 – SRDE -Status, Retrieve Detail_Route or Detail_State Error`

Validation rule: Retrieve path with detailed route option - No path ID associated to ONNS connection

Error code: `E282 – SCSI – Status, Conn State Invalid for this Operation`

Error response E282 defined for **RTRV-PATH** applied with *RetrieveDetail_Route* option shall also be valid in case of **RTRV-PATH** applied with *RetrieveDetail_State* option.

Validation rule: Retrieve path with detailed state option - Operation Invalid for Protection Type MBYNPRT

All error responses defined for **RTRV-PATH** applied with RetrieveDetail_Route option shall also be valid in case of **RTRV-PATH** applied with RetrieveDetail_State option.

Error code:
`E283 – SOPI – Status, Operation Invalid for Protection Type MBYNPRT`

For connections of protection type MBYNPRT the retrieve detailed state option does not provide any added value for the following reasons:

- If the connection is in STBY state, both ends of its associated path are not connected in the A- and Z-node and thus only FAILED signals can be monitored for the whole path throughout the domain irr. of any failures that may exist.
- If the connection is in USED state, its associated path is allocated by an MBYNWRK connection and can be monitored there.

Validation rule: Retrieve path - Invalid connection ID for group retrieval

If the source node Network Element receives a TL1 command containing a connID with tributary# = ALL, but the corresponding port is not a VCG, then one of the following error response is returned:

```
   srcTid 2001-06-01 08:00:00
M  ctag DENY
   IICG
   ″E050:13579468:\″Input, Invalid connection ID for connection
   group/connID″
;
```

The following list identifies conditions that will cause similar error responses specific to **RTRV-PATH** commands for groups of connections (including E050 above):

- Error code:
  `E050 – IICG – Input, Invalid connection ID for connection group/conn ID`
- Error code:
  `E051 – IIDG – Input, Invalid DetailRoute or DetailState option for Group retrieve`

Example: The command

```
RTRV-PATH:LT-SNNS9:snn:123456:::123.198.42.233&1-2-32-4-ALL:detailRoute=
   ON;
```

will produce the output E051.

The following error response will be raised in case connection alias and connection ID specified in the command do not match:

Error code: `E038 – IICS – Input, Inconsistent connAlias Specified`

Related TL1 messages

- "ED-PATH: Add or modify alias of ONNS connection" (p. 3-739)
- "ENT-PATH: Create ONNS path connection" (p. 3-1701)
- "DLT-PATH: Delete ONNS path connection" (p. 3-222)
- "REPT-PTHEVT notification" (p. 3-2175)
- "REPT-PTHCHG notification" (p. 3-2149)
- "REPT-NNCHG notification" (p. 3-2137)
- "ED-NNCNFG: Modify ONNS system configuration" (p. 3-450)
- "RTRV-NNCNFG: Retrieve ONNS configuration data" (p. 3-2555)
- "RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
- "RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
- "REPT-NNEVT notification" (p. 3-2144)
- "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
- "ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
- "DLT-CONN: Delete network connection" (p. 3-139)

Possible inconsistency during soak time

If the **RTRV-PATH** request with *detailstate* = on is started while the soak timer is running, the retrieved results may be inconsistent.

After the soak timer ends, the results are correct.

RTRV-PM: Retrieve performance monitoring

The **RTRV-PM** command is initiated by a user to request the NE to send the current and/or historical Performance Monitoring (PM) data associated with one or more facilities. For each value in the *montype* parameter of the **RTRV-PM** input parameters section, PM counter values will be stored in the current 15-minute period, the 32 previous 15-minute periods (past 8 hours worth of data), current day, and previous day periods.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-PM-*modifier*:*tid*:*aid*:*ctag*::[*montype*], [*monlev*], [*locn*],,*tmper*, [*mondat*], [*montm*];

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the port type or tributary on which the **RTRV-PM** command is to act on.
   Type: Alphanumeric value set.
   Range: The following values are for SONET only: STS1, STS3C, STS12C, STS48C, STS192C, OC3, OC12, OC48, OC48T, OC192, OC768, T3.
   The following values are for SDH only: VC3, VC4, VC44C, VC416C, VC464C, STM1E, STM1, STM4, STM16, STM16T, STM64, STM256, ODU1.
   The following value is for OTN only: OTU2
   Although the ODU layer is independent of SONET/SDH, the ODU1 modifier is listed here in the SDH section. This is because the ODU1 function is used for the Transparency feature, which is only supported for SDH.
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
   Type: AID
   Range: For STM<N>, STM<N>T, OC<N>, EC1 and OC<N>T modifiers: SPortAID.
   For T3 modifier: APortAID
   For OTU modifier: OTNPortAID
   For ODU<N> modifiers: VCGAID

For VC3, VC4, VC44C, VC416C, VC464C, STS1, STS3C, STS12C, STS48C and STS192C modifiers only: STS1NumAID.
And for modifiers VC4 and STS1: VCGNumAID.
Range is supported up to shelf level or system level; i.e.
ALL-ALL-#-#-ALL-ALL (on port level).
ALL-ALL-#-#-ALL-ALL-ALL (on tributary level).

4. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).

5. **montype** – Monitored parameter type
   This is the monitored parameter type for which data is requested. If the value is omitted, this parameter defaults to ALL.
   Type: Value set.
   Range: The following values are for all OC<N> and the EC1 modifiers: CVS, ESS, SESS, SEFS-S, LOSS-S, CVL, ESL, SESL, UASL, AISS-L, FC-L, CV-LFE, ES-LFE, SES-LFE, UAS-LFE, FC-LFE.
   The following values are for all OC<N>T modifiers: CVS, ESS, SESS, SEFS-S, LOSS-S.
   The following values are for all OC<N>T modifiers only: CVS-EGR, ESS-EGR, SESS-EGR, SEFS-S-EGR.
   The following values are also for OC192 and OC768 modifiers only: FECC-L.
   The following values are for STS1 (path) and are also applicable to STS3C, STS12C, STS48C and STS192C modifiers only: CVP, ESP, SESP, UASP, FC-P, CV-PFE, ES-PFE, SES-PFE, UAS-PFE, FC-PFE, PPJC-PGEN, NPJC-PGEN, PPJC-PDET, NPJC-PDET
   The following values are for all STM<N> modifiers: RS-N-ES, RS-N-SES, RS-N-BBE, RS-N-UAS, MS-N-ES, MS-N-SES, MS-N-BBE, MS-N-UAS, MS-F-ES, MS-F-SES, MS-F-BBE, MS-F-UAS.
   The following values are for all STM<N>T modifiers: RS-N-ES, RS-N-SES, RS-N-BBE, RS-N-UAS
   The following values are for all STM<N>T modifiers only: RS-N-ES-EGR, RS-N-SES-EGR, RS-N-BBE-EGR, RS-N-UAS-EGR
   The following values also are for STM64 and STM256 modifiers only: MS-N-FECC.
   The following values are for T3 modifier: CVL, ESL, ESA-L, ESB-L, SESL, LOSS-L, CVP, ESP, ESA-P, ESB-P, SESP, UASP, FC-P, SEFS-P, SAS-P, AISS-P, CV-PFE, ES-PFE, ESA-PFE, ESB-PFE, SES-PFE, UAS-PFE, FC-PFE, SAS-PFE, CVP-EGR, ESP-EGR, ESA-P-EGR, ESB-P-EGR, SESP-EGR, UASP-EGR, FC-P-EGR, SEFS-P-EGR, SAS-P-EGR, AISS-P-EGR, CV-PFE-EGR, ES-PFE-EGR, ESA-PFE-EGR, ESB-PFE-EGR, SES-PFE-EGR, UAS-PFE-EGR, FC-PFE-EGR, SAS-PFE-EGR
   The following values are for all ODU<N> modifiers only: ODU-N-ES, ODU-N-SES, ODU-N-BBE, ODU-N-UAS, ODU-F-ES, ODU-F-SES, ODU-F-BBE, ODU-F-UAS
   The following values are for VC3, VC4, VC44C, VC416C and VC464C modifiers

only: VC-N-ES, VC-N-SES, VC-N-BBE, VC-N-UAS, VC-F-ES, VC-F-SES, VC-F-BBE, VC-F-UAS, PJCPOS, PJCNEG
The following values are for OTU modifiers only: OTN-LOSS, OTN-FECC
The following value is also allowed for all input modifiers: ALL.
Default: ALL.

6. **`monlev`** – Monitored Level
   This parameter provides the level for the monitored PM parameter. The parameter is only evaluated if the value of *montype* parameter is ALL.
   Type: value set.
   Range:
   - 1-UP: Means the NE will report all non-zero data and all data which is marked as non-valid (parameter validity).
   - 0-UP: Means the NE will report zero and all positive data.

   Default: 1-UP.
7. **`locn`** – Location
   Location for which the PM data is monitored. The parameter value is only considered if the *montype* value is ALL; otherwise it is ignored for the calculation of the command reply (i.e has no impact on the command).
   Type: value set.
   Range:
   - NEND: Near-End PM data is requested.
   - FEND: Far-End PM data is requested.
   - ALL: All near-end and far-end data is requested. It applies only to input direction.

   Default: ALL.
8. **`tmper`** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.
9. **`mondat`** – Monitored Date
   This requests the beginning date of the interval for which the PM data is to be reported. For both values of tmper parameter, the maximum allowable range of valid dates includes the current and previous one day.
   Type: value set.
   Range:
   - MM-DD: Month-Day. A leading zero (first M or first D) is optional and may be omitted.
   - ALL: Current and previous day.

   Default: Current date

10. **`montm`** – Monitored Time

    This parameter specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. The user is able to specify a monitored time in three modes:

    a. A monitored time interval (i.e., from hour-minutes to hour-minutes: HOD-MOH&&THOD-TMOH).

    b. A single time interval (i.e., the hour and minutes for the appropriate 15 minute data: HOD-MOH).

    c. All applicable beginning times for the given *mondat* value (i.e., all).

    Where HOD-MOH represents the start of the requested PM retrieval period (hours and minutes) and THOD-TMOH represents the end of the requested PM retrieval period. The value ALL must be used if the *mondat* value provided is ALL.
    Type: Character string.

    Range:

    - HOD-MOH: This specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. This parameter may be omitted or have one of the following values: Hour Of Day- Minute Of Hour, where HOD ranges from 0 to 23 and MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.
    - THOD-TMOH: To Hour Of Day-To Minute Of Hour, where THOD ranges from 0 to 23 and TMOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0. Parameter grouping is used when specifying THOD-TMOH. For example, 08-15&&09-00 specifies the HOD-MOH and THOD-TMOH of 08-15 and 09-00, respectively. THOD-TMOH of $$-$$indicates the current Hour Of Day-Minute Of Hour. THOD-TMOH of ##-##indicates the previous 15-min interval.
    - ALL: All applicable beginning times for the given *mondat* value.

    If the *tmper* value is 1-DAY, the *montm* parameter will not be used.
    Default:
    If THOD-TMOH is not specified, only the PM data for the retrieval period specified by HOD-MOH will be retrieved.
    If HOD-MOH is omitted, the PM period for the current time will be reported.

Output format

If the NE fully complies with the RTRV-PM request, the following normal completion response is returned:

```
    sid date time
M  ctag COMPLD
   ″aid:montype,monval,[validity],[tca],[locn],,tmper,[mondat],[montm]″
   . . . . .
   . . . . .
   . . . . .
;
```

Note: The trailing comma has to be present even if the optional parameter is not supported for a dedicated release.

The output parameters *sid, date, time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Normal completion message – empty reply

If there is no PM data stored within the NE, then the following response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

1. **aid** – Access Identifier
   Description: Access identifier.
   Type: AID
   Range: For STM<N>, STM<N>T, OC<N>, EC1 and OC<N>T modifiers only: SPortAID.
   For T3 modifier: APortAID
   For OTU modifier: OTNPortAID
   For ODU<N> modifiers only: VCGAID
   For VC3, VC4, VC44C, VC416C, VC464C, STS1, STS3C, STS12C, STS48C and STS192C modifiers only: STS1NumAID.
   And for modifiers VC4, VC464C, STS1, STS3C and STS192C: VCGNumAID
2. **montype** – Monitored Parameter Type
   See montype in the input parameters section.
3. **monval** – Monitored Value
   This parameter specifies the measured value of the parameter specified in montype.
   Type: Integer.
   Range: 0-4294967295
   The specified range is the maximal range taking all PM parameters into consideration. Specific PM parameters may have a smaller range.

   W.r.t. value range for the specific PM parameters see

   - Section "Output parameters" in "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982) (if thresholding is applicable for that parameter)
   - PM parameter introduction section (if thresholding is not applicable for that parameter).

4. **`validity`** – validity indicator
   This parameter indicates the validity of the PM data.
   Type: Integer.
   Range: 1: valid; (omitted): non-valid.
5. **`tca`** – Threshold Crossing Alert
   Type: value set.
   Range for the current interval:
   - in TCA mode TR, that the current value exceeds the provisioned threshold
   - in TCA mode TR-RTR, that a *tca* occurred since the beginning of this; interval or that the tca reset condition was not fulfilled in any previous; interval since the last tca occurred

   f the value is omitted, it indicates
   - in TCA mode TR, that the current value is below the provisioned threshold
   - in TCA mode TR-RTR, that no tca occurred since the beginning of this; interval and that the tca reset condition was fulfilled at least in one previous , interval

   For previous intervals, the tca parameter indicates the state at the end of the interval.
6. **`locn`** – Location
   Location for which the PM data is monitored.
   Type: value set.
   Range:
   - NEND: Near-End PM data is requested
   - FEND: Far-End PM data is requested.
7. **`tmper`** – Time Period
   See in the input parameters section.
8. **`mondat`** – Monitored Date
   This parameter specifies the beginning date of the interval for which the PM data is reported. The parameter can be omitted in case of current date.
   Type: value set.
   Range: MM-DD: Month-Day
   Default: Current date
9. **`montm`** – Monitored Time
   This parameter specifies the beginning time of day for the reported pm register. The parameter is optional and may be omitted for the reporting of current bins (registers).
   Type: Character string.
   Range: HOD-MOH: HOD ranges from 0 to 23; MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.

Example input/output

The following example shows the successful completion of a **RTRV-PM** command by the NE for a VC3:

```
RTRV-PM-vc3:UNITE-NE:1-1-#-#-5-1-27:123456::vc-f-es,1-up,,,15-min ,,
00-00&&08-00;
   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
   ”1-1-#-#-5-1-27:vc-f-es,17,,tca,,,15-min,,00-00”
   ”1-1-#-#-5-1-27:vc-f-es,10,,tca,,,15-min,,00-15”
   . . .
   ”1-1-#-#-5-1-27:vc-f-es,15,,tca,,,15-min,,08-00”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-PM** command. See also subsection ’Partial Completion on RTRV Commands’ w.r.t. handling of internal error (e.g. communication to XC320 fails) for a subset of replies.

Related TL1 messages

- “RTRV-EPM: Ethernet performance monitoring” (p. 3-2352)
- “DLT-TCA-PROF: Delete TCA profile” (p. 3-259)
- “ED-TCA-PROF: Edit TCA profile” (p. 3-1214)
- “ENT-TCA-PROF: Enter TCA Profile” (p. 3-1894)
- “RTRV-TCA-PROF: Retrieve TCA profile” (p. 3-2982)

References

GR-833-CORE

RTRV-PREDEFRAG: Perform an ONNS fragmentation analysis

Overview

The **`RTRV-PREDEFRAG`** command is used to perform an analysis about the fragmentation status of a link or other entity.

This command is used by an operator in preparation for doing a defrag operation on a port or between multiple ports in a link between two nodes.

The command causes the receiving node to do an analysis of the level of fragmentation that currently exists on the specified port or link. An analysis is performed of how much usable bandwidth will be gained if defragmetation were to be done, and the time slot structure as it would be following a defragmentation operation is computed. Note that all this is done in an analysis mode. No real defragmentation is done until a subsequent “operate defrag” command is issued. The rtrv-predefrag command causes the network element to send the results of the defrag analysis in a status report back to the management system in a TL1 response.

As above stated the RTRV-PREDEFRAG is done in an analysis mode and does not guarantee the same result for Gain and *PostLDGrouping* after real defragmentation.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Abortable: No.

Input format

Command syntax: **`RTRV-PREDEFRAG:`***srcTid:srcAid:ctag::scope:targetID*;

Input parameters

Command parameters

1. **`srcTid`**
   Target Identifier of the source node is the logical name of the Host NE. ONNS uses the same *tid* used by Host NE.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: The *srcTid* parameter is a case insensitive ASCII character string of up to 20 characters with the allowed character set as described above.
2. **`ctag`** – Correlation Tag
   This parameter is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   The *ctag* parameter is a character strings of up to 6 ASCII characters. The allowed ASCII character string is comprised of alphanumeric characters beginning with a letter, or decimal numerals (a decimal numeral is a string of decimal digits with an optional non-trailing ".").
3. **`srcAid`**
   Source node ONNS application access identifier of the ONNS controller OS command interface port.
   Range: Source node Access IDentifier parameter of the source node controller ONNS command interface entity has the allowed values shown here: snn.
4. **`scope`**
   This parameter specifies the scope of the defrag operation: intra-port or inter-port (intra link).

   Range

   - PORT: specifies the defrag is intra-fiber operation on a specific port. In this case *targetID* specifies the local port id same as the *portAID*. The corresponding neighbor node and neighbor *portAID* can be deduced from the *targetID*.
   - LINK: specifies the link defrag is inter fiber but within a link bundle. Also, the link defrag operation is performed on all the bundles between the two neighboring nodes. TargetID specifies the neighbor nodes ASTN node ID.
5. **`targetID`**
   The specification of the *targetID* depends on the setting of the parameter scope: PORT specifies the AID of the port on the local node that is to be defragmented;LINK specifies the ASTN node ID of the neighbor node of the link that is to be defragmented.
   Type: Four dash separated numeric string for the portAID. Four dot separated numeric string for the neighbor ASTN node ID.
   Range. Each numeric string: 0…255.

Output format

If the RTRV-PREDEFRAG request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″PreLDGrouping=NuM&NaM&....
    PostLDGrouping=NuM&NuM&...
    Gain=NSTSn&..″
;
```

The pre and post defrag time slot structure is indicated by two strings consisting of ″&″ separated substrings. The substrings have the form NuM or NaM where N is a number indicating the number of STS1s that make up that slot and M indicates the rate (3C, 12C etc..). The letters ″u″ and ″a″ indicate whether the time slot is used or is available respectively. The value of M while specifying the number of available STS1s is always blank. Thus 2a means two STS1 slots are available.

The gain parameter indicates number of additional time slots that will become available once defrag is done.The highest rate time slots are indicated first, followed by the lower rates, with each signal rate substring being separated by an &. The fractional STS1s that form the remainder are not indicated. If there is nothing to be gained from the defrag operation, the gain parameter will say NONE.

Normal system response to RTRV-PREDEFRAG for inter-port (link) defragmentation

If the RTRV-PREDEFRAG request completes successfully, the following normal completion response is returned:

```
   srcTid date time
M  ctag COMPLD
   ″BundleID=bundlID_1,Gain=NSTSn&...″
   ″PortID=portID_1,
    PreLDGrouping=NuM&Na&...
    PostLDGrouping=NuM&Na&...″
   ...
   ″PortID=portID_m,
    PreLDGrouping=NuM&Na&...
    PostLDGrouping=NuM&Na&...″
   ...
   ″BundlID=bundleID_m,Gain=NSTSn&...″
   ″PortID=portID_1,
    PreLDGrouping=NuM&Na&...
    PostLDGrouping=NuM&Na&...″
   ...
   ″PortID=portID_m,
    PreLDGrouping=NuM&Na&...
```

```
    PostLDGrouping=NuM&Na&..."
;
```

The pre and post defrag time slot structure is indicated by two strings consisting of "&" separated substrings. The substrings have the form NuM or Na where N is a number indicating the number of STS1s that make up that slot and M indicates the rate (3C, 12C etc..). The letters "u" and "a" indicate whether the time slot is used or is available respectively. The available timeslots are ever shown in STS1s and no additional rate value M is needed. Thus 2a means two STS1 slots are available.

The gain parameter indicates number of additional time slots that will become available once defrag is done.The highest rate time slots are indicated first, followed by the lower rates, with each signal rate substring being separated by an &. The fractional STS1s that form the remainder are not indicated. If there is nothing to be gained from the defrag operation, the gain parameter will say NONE.

The inter-port defragmentation record will be send in data blocks that are less than 4096 characters. A data block will end with the pre and post defrag timeslot structure. On all messages but the last, a RTRV completion code is used and the semicolon (;) is replaced with a (>) terminator. Then, on the last record block, the COMPLD completion code is used with the final semicolon (;) terminator.

Output parameters

The output parameters *srcTid*, *date*, *time*, *ctag*, *completion code* and their syntax are included in the section "Output parameters" of the "RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698) command.

1. **`bundleID`**
   A unique numeric ID assigned to the link bundle on this node.
   Type: Integer
   Range: >0
2. **`portID`**
   Port ID for which the NNDR data is retrieved. The portID consists of Bay/Frame #, Shelf #, Slot/Pack #, Port # .
   Type: dash-separated list of 4 numeric strings
   Range: Bay/Frame #1, Shelf #1, Slot/Pack #1,..39 Port #1..16,
3. **`PreLDGrouping`**
   An ordered & separated string of numbers indicating the availability or unavailability of a group of time slots of different rates. The unit of measure is STS1. Availability is indicated by "a" and unavailability (used) by "u". Thus 1u3 indicates that one STS3c slot is used up, and 2a indicates that two STS1 slots are available. Notice that the available timeslots are always specified in terms of the number of STS1s. Hence, the "1" at the end is understood and not included to conserve space.

Type: &-separated list of numbers.
Example for an OC12 port:
PreLDGrouping=1u3&2a&1u1&1u3&1u1,2a, indicates one used STS3c, two available STS1, one used STS1, one used STS3c, 1 used STS1, two available STS1.
The size of the output string will vary depending on whether the port is OC3, OC12, OC48, OC192, OC768 and also on the distribution of the used/available time slots. For an OC192 port, the output string would indicate the STS48c, STS12c, STS3c and STS1 slots used up and the remaining unused STS1 slots.

4. **`PostLDGrouping`**
   An ordered string formatted the same way as the *PreLinkDefragGrouping* string, indicating the time slot structure once the defrag operation is completed.
   Type: &-separated list of numbers
   Example for the above case of the OC12 port, the PostLDGrouping=2u3&2u1&4a, indicating that two STS3c slots are used, two STS1 are used and four STS1 slots are available. Thus another STS3c can be accommodated.

5. **`Gain`**
   A parameter indicating whether the defrag will result in an improvement or not. ″NONE″ indicates no improvement, any other value indicates the ″delta″ improvement over the existing time slot structure. The delta is computed in terms of the additional signals that can be accommodated as a result of the defrag starting with the highest signal rate possible for that port. The improvement is represented as a string composed of one or more of the following substrings: ″STS3c″, ″STS12c″, ″STS48c″, ″STS192c″. Each substring has a numeric string preceding it indicating the number of timeslots of that type that will be available once defrag is performed. The highest signal rate is stated first, then the next highest rate and so on.
   Type: &-separated list of numbers concatenated with value set or the special value NONE.
   Range: The value is ″NONE″ if no gain. Else it indicates the additional timeslot that are now available. In the above example: Gain=1STS3C. Gain will be an & separated list if there are multiple gains. E.g: 1STS12c& 3STS3c would mean that 1 STS12c and 3 STS3c connections could be accommodated on that port once the defrag is done. Note only the additional available timeslots are included in the response.

Example input/output

Input/output examples for RTRV-PREDEFRAG intra-port defragmentation

The following example shows a successful completion of a **`RTRV-PREDEFRAG`** command:

```
RTRV-PREDEFRAG:NN001:snn:ed01::PORT,1-1-2-1;

   NN001 2002-01-12 01:03:16
M  ctag COMPLD
```

```
   ″PreLDGrouping=1u3&2a&1u1&1u3&1u1&2a
    PostLDGrouping=1u3&3a&1u3&2u1&1a
    Gain=1STS3C″
;
```

Input/output examples for RTRV-PREDEFRAG inter-port defragmentation

The following example shows a successful completion of a **RTRV-PREDEFRAG** command:

```
RTRV-PREDEFRAG:NN001:snn:ed01::Link,29.13.80.192;

   NN001 2002-01-12 01:03:16
M  ctag COMPLD
   ″BundleID=1,Gain=2STS3C″
   ″PortID=1-1-2-1
    PreLDGrouping=1u3&2a&1u1&1u3&1u1&2a
    PostLDGrouping=1u3&3a&1u3&2u1&1a″
   ″PortID=1-1-2-2
    PreLDGrouping=1u3&2a&1u1&1u3&1u1&2a
    PostLDGrouping=1u3&3a&1u3&2u1&1a″
;
```

Error responses

Invalid *PortID*

If the NE receives an RTRV-PREDEFRAG command with an *PortID* parameter value that is invalid the following error response is returned:

Error code: `E410 – Input, Invalid PortID`

Invalid *ASTNNodeID*

If the NE receives an RTRV-PREDEFRAG command with an *ASTNNodeID* parameter value that is invalid the following error response is returned.

Error code: `E512 – Input, Invalid Neighbor node ID`

Related TL1 message

- "OPR-DEFRAG: Perform defragmentation " (p. 3-2028)

RTRV-PRMTR-DATA: Retrieve parameter data

The **RTRV-PRMTR-DATA** command retrieves information about the databases being in NVM. It identifies the databases' release numbers and provides also other information such as their states.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 2

Abortable: No

Input format

Command syntax: **RTRV-PRMTR-DATA**:*tid*::*ctag*::*[amount]*;

Input parameters

Command parameters

1. **tid**- Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **amount** – Amount Of Data To Return
   This parameter specifies whether the current generic or previous generic or both is returned. If a value is omitted, then the current (active) generic is returned.
   Type: Alphanumeric value set.
   Range: CURRENT (default), PREVIOUS, ALL.
   Default: CURRENT.

Output format

The following normal completion response is returned, if the RTRV-PRMTR-DATA request completes successfully:

```
   sid date time
M  ctag COMPLD
   "nvm:spec_block"
       ...
       ...
   "nvm:spec_block"
;
```

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response. The output parameters listed there also apply to the **RTRV-PRMTR-DATA** command. Additional parameters that specifically apply to this command are defined in the following:

1. **nvm** – Non Volatile Memory Identifier
   This parameter specifies the region of non volatile memory data (current or previous).
   Type: Value set.
   Range:
   - NVM-A: for current NVM Data
   - NVM-B: for previous NVM-Data.

   Default: Only current NVM data is reported except for *amount* having been specified to ALL.

Spec block parameters

This parameter provides information about the stored database. It contains the parameters *release, size, backuptime, restoretime, backupfile, restorefile, priorbackuptime*, *priorbackupfile*, *priorrestoretime*, *priorrestorefile*, *state*, *tid* and *serialnumber*.

When software is being downloaded to previous NVM, the database parameters for NVM-B do not need initialization because they still reflect information pertaining to the currently executing generic.

During installation, the following parameters are omitted (initialized to a blank field) for NVM-A:

backuptime, restoretime, backupfile, restorefile, priorbackuptime, prioirrestoretime, priorbackupfile, priorrestorefile.

When a backup is performed for the database in current NVM, the values of *restoretime*, *restorefile*, *priorbackuptime*, *priorbackupfile* is not modified.

1. **`release`** – Release Number
   This parameter represents the release number contained in the file. The parameter will be omitted if state is ’INITIAL’.
   Type: value set of the form x.y.z
   Range: x, y, z are integer numbers from 00 to 99 in two-digit form, that means leading zeros are not suppressed. x is not 00.
2. **`size`** – Size Of Allocated NVM
   This parameter indicates the allocated NVM size in bytes.
   It is not dynamically updated within the generic until data migration or software upgrade which may change the budget of NVM space used for database files.
   Type: Numeric string.
   Range: 0 to 99999999
   Default: In case the software generic on the NVM is empty, the value 0 is reported.
3. **`backuptime`**
   This parameter provides time and date when the last backup to a remote file store has been finished. It is omitted if the system has never been backed up.
   The NE updates *backuptime* at the OS just before file transfer starts and then update the value of current NVM’s *backuptime* when the backup finishes.
   This implies the value of *backuptime* contained in backupfile at the OS will represent a slightly earlier time than the value contained in current NVM’s database.
   backuptime is updated whether or not the file transfer was successful.
   Type: value set of the format: YY-MM-DD HH-MM-SS.
   Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.
4. **`restoretime`**
   This parameter reflects the time and date when the last restore from a remote file store has been finished. It is omitted, if the system has never been restored.
   When a backup is performed for the database in current NVM, the values of *restoretime* (in current NVM, previous NVM) are not modified.
   Only database files at the current or previous NVMs shall have values for *restoretime*.
   Type: value set of the format: YY-MM-DD HH-MM-SS
   Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.

5. **`backupfile`**
   This parameter indicates the remote location where the last backup is stored. Omitted if never backed up.
   The NE will update *backupfile* before the file transfer starts.
   Thus, both the source (current NVM) and the destination (*Navis*® Optical EMS or CIT) will have the same value for *backupfile*.
   Type: text string.
   Range: The parameter has one of the following formats:
   - *″CIT,filename″*
   - *″EMS,filename″*
   - *″FTTD″,FTP URL″*

   The syntax contained in *filename* should be consistent with the file system. For a DOS file system, back slashes (\) should be used. For a UNIX file-system, forward slashes (/) should be used.
   Because a (\) is interpreted as the escape character, it will be represented in the output reply as a double backslash (\\).
   The FTP URL must be in the format:
   ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>
   - <login>: default value: ″anonymous″
   - <password>: default for <login> ″anonymous″ is ″″ (empty). For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
   - <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0..255.
   - <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
   - <file_name>: the complete path name of the file.
6. **`restorefile`**
   This parameter reflects the remote location where the last restore came from. Omitted if never restored.
   When a backup is performed for the database in current NVM, the values of *restorefile* (in current NVM, previous NVM, the destination) will not be modified.
   restorefile is updated whether or not the file transfer was successful.
   Type: text string.

Range: *restorefile* has one of the following formats:

- *″CIT,filename″*
- *″EMS,filename″*
- *″FTTD″,FTP URL″*

The syntax contained in *filename* should be consistent with the file system. For a DOS file system, back slashes (\) should be used. For a UNIX file-system, forward slashes (/) should be used.
Because a (\) is interpreted as the escape character, it will be represented in the output as a double backslash (\\).
The FTP URL must be in the format:
ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>

- <login>: default value: ″anonymous″
- <password>: default for <login> ″anonymous″ is ″″ (empty). For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
- <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0..255.
- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

7. **`priorbackuptime`**
This parameter gives the time and date when the next-to-last backup to a remote file store finished. It is omitted, if there is no next-to-last-backup.
When a backup is performed for the database in current NVM, the values of *priorbackuptime* (in current NVM, previous NVM, the destination) will not be modified.
Type: value set of the format: YY-MM-DD HH-MM-SS
Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.
Default: n. a.

8. **`priorrestoretime`**
This parameter specifies the time and date when the next-to-last restore from a remote file store finished. It is omitted, if there is no next-to-last restore.
Type: value set of the format: YY-MM-DD HH-MM-SS.
Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.

9. **`priorbackupfile`**

This parameter indicates the remote location where the next-to-last backup is stored. It is omitted, if there is no next-to-last backup.
When a backup is performed for the database in current NVM, the values of *priorbackupfile* (in current NVM, previous NVM, the destination) will not be modified.
Type: text string

Range: *priorbackupfile* has one of the following formats:

- *″CIT,filename″*
- *″EMS,filename″*
- *″FTTD″,FTP URL″*

The syntax contained in *filename* should be consistent with the file system. For a DOS file system, back slashes (\) should be used. For a UNIX file system, forward slashes (/) should be used.
Because a (\) is interpreted as the escape character, it will be represented in the output reply as a double backslash (\\).

The FTP URL must be in the format:
ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>

- <login>: default value: ″anonymous″
- <password>: default for <login> ″anonymous″ is ″″ (empty).For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
- <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0..255.
- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

10. **`priorrestorefile`**

This parameter reflects the remote location where the next-to-last restore came from. It is omitted, if the is no next-to-last restore.
Type: text string

Range: *priorrestorefile* has one of the following formats:

- *″CIT,filename″*
- *″EMS,filename″*
- *″FTTD″,FTP URL″*

The syntax contained in *filename* should be consistent with the file system. For a DOS file system, back slashes (\) should be used. For a UNIX file-system, forward slashes (/) should be used.
Because a (\) is interpreted as the escape character, it will be represented in the output reply as a double backslash (\\).

The FTP URL must be in the format:
ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>

- <login>: default value: ″anonymous″
- <password>: default for <login> ″anonymous″ is ″″ (empty).For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
- <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0..255.
- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

11. **`state`** – State of the Database

This parameter describes the state of the database per NVM region:

- CORRUPT (current, previous): Unavailable for invocation or execution (e.g. corrupted check sum)
- MISMATCH (previous): Contained ES64U databases do not fit to the actual configuration
- DOWNLOADED (previous): Applies to the previous bin of database. Database has been downloaded and is ready for activation.
- INITIAL (current): Nothing in NVM.
- INITIAL (previous): Nothing in previous partition of NVM.
- INPROGRESS (current): Backup or restore in progress.
- INPROGRESS (previous): Restore is in progress.
- PREVIOUS (previous): A previously active database.
- PRESERVED (previous): Previously active database during trial state
- RUNNING (current): The currently active database.

The normal database states of current NVM are: RUNNING and INPROGRESS. Most of the time its state is equal to RUNNING, signifying that it contains a copy of the database that is currently in use. When a DB backup or a DB restore makes the NVM inaccessible, current NVM will have a database state of INPROGRESS. The normal database states of previous NVM are: PREVIOUS and INPROGRESS. Most of the time its state is equal to PREVIOUS, signifying that it contains a copy of the database that was used with the previous generic. The state PRESERVED indicates a copy of the previously active database after application of a new SW generic, while the new SW generic is in trial phase. During the trial phase a switch back to the previously installed SW generic and the previously active database is possible. When a restore makes the NVM inaccessible, previous NVM will have a database state of INPROGRESS.

The state DOWNLOADED applies to the two-step DB restoration process in which

a DB first is being downloaded and at a separate step being activated.
If an NVM has never contained a database, then the RTRV-PRMTR-DATA output for that NVM will consist of *state* INITIAL and size 0.
If an NVM is failed or missing, then RTRV-PRMTR-DATA will report only the state output variable for that NVM. *state* will have the value CORRUPT.

If an NVM contains a database generic, which contains ES64U databases and

a. the CTL does not support the ES64U feature or

b. the XC does not support the ES64U feature

then the *state* shall have the value CORRUPT
If an NVM contains a database generic, which contains ES64U databases but the contained ES64U databases do not fit to the actual ES64U configuration then the *state* shall have the value MISMATCH
Type: Alphanumeric value set.
Range: CORRUPT, MISMATCH, DOWNLOADED, INITIAL, INPROGRESS, PREVIOUS, PRESERVED, RUNNING.
Default: INITIAL

12. **`tid`** – target identifier
This parameter gives the TID that was stored in this NVM. The parameter will be omitted if state is INITIAL.
This parameter is not evaluated in the NE. *serialnumber* will be used to check whether database belongs to correct NE.
See "RTRV-HDR: Retrieve header" (p. 3-2417) for allowable tid values and details.

13. **`serialnumber`** – serial number
This parameter gives the NE serial number from the database control information showing to which NE the database belongs to. Comparing this serial number with the actual NE serial number from the backplane EEPROM may help troubleshooting when being in maintenance mode. The parameter will be omitted if state is INITIAL.
Type: Alphanumeric value set.
Range: 12 times the space character or the 12 byte serial number of electronic label.
Default: 12 times the space character.

Example input/output

In the following example for the **`RTRV-PRMTR-DATA`** command, current NVM contains a copy of the RUNNING database. The database was last backuped at 9:00 and the backup was to CIT:

```
RTRV-PRMTR-DATA:LT-UNITE::123456;
   LT-UNITE 00-11-01 09:45:28
M  123456 COMPLD
   "NVM-A:release=01.02.04,size=8234567,backuptime=00-11-01 09-00-
      00,backupfile=\"CIT,\\data\\00_11_01B\",priorbackuptime=00-11-01 08-00-
      00,priorbackupfile=\"CIT,\\data\\00_11_01A\",state=running,tid=LT-
      UNITE,serialnumber=01NB12345678"
```

;

Related TL1 messages

- "APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
- "CPY-MEM: Copy memory" (p. 3-114)
- "RTRV-PRMTR-SFTWR: Retrieve parameter software" (p. 3-2753)

RTRV-PRMTR-SFTWR: Retrieve parameter software

The **RTRV-PRMTR-SFTWR** command retrieves information about the software generics being in NVM. It identifies the generics' release number and provides also other information such as its state.

RTRV-PRMTR-SFTWR can be used to determine what generic is currently executing. It can also be used to determine what generic was previously executing. If a generic was downloaded but not installed, then the output describes the downloaded generic instead of the previously executing generic.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 2

Abortable: No

Input format

Command syntax: **RTRV-PRMTR-SFTWR**:*tid*::*ctag*::*[amount]*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **amount** – Amount Of Data To Return
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the NE fully complies with the **RTRV-PRMTR-SFTWR** command and the *amount* request is specified to all, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″nvm:spec_block″
        ...
        ...
   ″nvm:spec_block″
;
```

Output parameters

Refer to the "Output parameters" section of the "RTRV-HDR: Retrieve header" (p. 3-2417) command for a normal completion response. The output parameters listed there also apply to the **RTRV-PRMTR-SFTWR** command. Additional parameters that specifically apply to this command are defined in the following:

1. **nvm** – non volatile memory identifier
   This parameter specifies the version of NVM software (current or previous).
   Type: value set.
   Range: NVM-A for Current NVM-Software, NVM-B for Previous NVM-Software.
   Default: Only current NVM software is reported except for *amount* having been specified to ALL.

Spec block parameters

This parameter provides information about the stored software. It contains the parameters *release, size, supplier, type, delivered, installed, path, state.*

1. **release** – Release Number
   Type: value set of the form x.y.z
   Range: x, y, z are integer numbers from 00 to 99 in two-digit form, that means a leading zeros are not suppressed. x is not 00.
   Default: 01.00.00.
2. **buildnmbr** – Build Number
   This parameter identifies the software load contained in the file. It is an internal representation for test purpose.
   In contrast to the *release* parameter, which is the same for all software loads of a product release, the *buildnmbr* differs for all produced software loads, for example, produced for test purposes. The build number is reported for the software load in the current NVM. For the previous NVM the default value is reported.

   The *buildnmbr* typically contains the following informations to identify the specific software load:

   - internal load number (SUSI number)
   - internal build number of the load

   Type: Text string
   Range: Up to 20 characters.
   Default: UNKNOWN.
3. **size** – Program Size
   This parameter indicates the software generic size in bytes.
   Type: numeric string.
   Range: 0 to 99999999.
   Default: In case the software generic on the NVM is empty, the value 0 is reported.

4. **`supplier`**
   This parameter provides the supplier of generic.
   Type: value set.
   Range: 0 to 128 characters
   Default: Alcatel-Lucent..
5. **`type`** – System Type
   This parameter specifies the type of system.
   Type: Value set.
   Range: WaveStar_UNITE
   Default: WaveStar_UNITE.
6. **`delivered`**
   This parameter specifies the time and date when the download finished.
   Type: value set of the format: YY-MM-DD HH-MM-SS.
   Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.
   Default: 00-01-01 00-00-00.
7. **`installed`**
   This parameter specifies date and time when the installation started.
   Type: value set of the format: YY-MM-DD HH-MM-SS
   Range: Y, M, D are numbers in two-digit format (i.e. leading zeros are not suppressed). YY represents the last two digits of the year, MM the month, DD the day of the month. H, M, S are numbers in two-digit format (i.e. leading zeros are not suppressed). HH represents the hour, MM the minutes, SS the seconds.
   Default: 00-01-01 00-00-00.
8. **`path`**
   This parameter specifies the location in a remote file system from where the SW Generic has been downloaded. In the case of a factory installed SW generic the parameter shall have the value “PRE-LOADED”.
   Type: text string.

   Range: The path parameter has one of the following formats (see description for details):

   - ″CIT,psel,ssel,tsel,nsap,filename″
   - ″EMS,psel,ssel,tsel,nsap,filename″
   - ″FTTD,psel,ssel,tsel,nsap,FTP URL″
   - ″PRE-LOADED″

   The sizes for *psel, ssel, tsel, nsap* are as follows:

   - *psel* (path selector): 0-4 bytes (0-8 hex characters)
   - *ssel* (section selector): 0-4 bytes (0-8 hex characters)

- *tsel* (transport selector): 0-16 bytes (0-32 hex characters)
- *nsap* (network service access point): 20 bytes (40 hex characters).

There can only be an even number of hex characters since they are generated from bytes and each byte corresponds to 2 hex characters.

From MSB (Most Significant Byte) to LSB (Least Significant Byte), these bytes are (the number in parenthesis equals the number of bytes): IDP(3), DFI(1), ORG(3), RES(2), RD(2), AREA(2), SYS(6), SEL(1). The SEL field of the *nsap* is also called the NSEL.

The syntax contained in *filename* should be consistent with the file system. For a DOS file system, back slashes (\) should be used. For a UNIX file-system, forward slashes (/) should be used.

Because a (\) is interpreted as the escape character, it will be represented in the RTRV-PRMTR-SFTWR output as a double backslash (\\).

When the CIT is used to enter a file name for download backup, or restore, the craft is not required to enter a double backslash; the CIT software will enter the extra backslash into the TL1 CPY-MEM command.

For FT-TD *psel*, *ssel*, *tsel* and *nsap* of the GNE is reported. The FTP URL contains the CIT or EMS and the filename.

The FTP URL must be in the format:
ftp://[<login>[:<password>]@]<host>[:<port>]/<file_name>

- <login>: default value: ″anonymous″
- <password>: default for <login> ″anonymous″ is ″″ (empty).For security reasons, the <password> portion of the FTP file format must be masked prior to generating any autonomous messages.
- <host>: the <host> portion is an internet address of the form a.b.c.d with a, b, c and d in the range of 0..255.
- <port>: this is the port number of the FTP server, given when deviating from the standard port number 21.
- <file_name>: the complete path name of the file.

9. **`state`**

This parameter describes the state of the generic per NVM region:

state (NVM):description

- CORRUPT (current, previous): Unavailable for invocation or execution.(e.g. corrupted check sum)
- DOWNLOADED (previous): Applies to the previous bin of database. Database just downloaded or ready for installation.
- INITIAL (current): Nothing in NVM.
- INITIAL (previous): Nothing in previous partition of NVM.
- INPROGRESS (current): Installation is in progress..
- INPROGRESS (previous): Previously executing generic. Available for installation

- RUNNING (current): Currently executing generic.
- PRESERVED (previous): Previously executing generic in trial phase
- PREVIOUS (previous): Previously executing generic.

The normal (as contrasted with CORRUPT) states of current NVM are: RUNNING and INPROGRESS. The state is usually equal to RUNNING, signifying that it contains a copy of the generic that is currently executing. When an installation is in progress, current NVM will have a state of INPROGRESS.

The normal states of previous NVM are: PREVIOUS, DOWNLOADED and INPROGRESS. INPROGRESS means that a download is in progress. When the download terminates successfully, the state transitions to generic is installed, the NVM pointer is adjusted, and the NVM partition then identified as previous NVM will have the state set to PREVIOUS. The state PRESERVED indicates the trial phase after application of a new SW generic. During the trial phase a switch back to the previously installed SW generic and the previously active database is possible.

If an NVM has never contained a software generic, then the RTRV-PRMTR-SFTWR output for that NVM will consist of State=INITIAL.

If an NVM is corrupt, failed or missing, the RTRV-PRMTR-SFTWR output for that NVM will consist only of a single line containing State=CORRUPT.

Type: Alphanumeric value set.

Range: CORRUPT, DOWNLOADED, INITIAL, INPROGRESS, PREVIOUS, PRESERVED, RUNNING.

Default: INITIAL.

Example input/output

In the following example for the **RTRV-PRMTR-SFTWR** command, current NVM has a copy of the executing program (normal condition), and previous NVM has a copy of the previous generic. Also, it was assumed, that the CIT is DOS based and *Navis*® Optical EMS UNIX based. This was done to illustrate the different formats for path. Notice that the current and previous NVM indicate the same number for free (and also unavailable) space since current and previous information is stored in the same physical NVM.

```
RTRV-PRMTR-SFTWR:LT-UNITE::123456::ALL;
   LT-UNITE 00-11-01 09:45:28
M  123456 COMPLD
   ″NVM-A:release=01.02.04,size=11423175,supplier=Lucent,type=Unite,delivered=00-11-
      02 08-00-10,installed=00-11-02 08-00-
      13,path=\″CIT,01,01,03,39840F8000000000000000000008006A11234500,\\unite\\generic\\1_2
   ″NVM-B:release=01.02.03,size=11343236,supplier=Lucent,type=Unite,delivered=00-11-
      01 18-30-10,installed=00-11-01 22-45-
      30,path=\″EMS,01,01,02,39840F8000000000000000000008006A11118500,/unite/generic/1_2_03
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-PRMTR-SFTWR** command.

Related TL1 messages

- "APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
- "CPY-MEM: Copy memory" (p. 3-114)
- "RTRV-PRMTR-DATA: Retrieve parameter data" (p. 3-2744)

RTRV-PROTN-GRP: Retrieve protection group

The **RTRV-PROTN-GRP** command retrieves protection group information for all existing protection group entities. The command can be used to retrieve also sets of protection groups by using ranges. The general format can be used to retrieve protection group information for different protection group types.

Input parameters for **RTRV-PROTN-GRP** identify the type of protection group that is of interest and also a specific member (or perhaps a range of members) of that protection group. If the specified member does not belong to a protection group, then the command returns COMPLD, but contains no information about the protection group. This means that the AID is unprotected.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

1+1 equipment protection group

Command syntax: **RTRV-PROTN-GRP**:*tid*:*aid*:*ctag*::*protype*;

multiplex section/line protection group

Command syntax: **RTRV-PROTN-GRP**:*tid*:*aid*:*ctag*::*protype*;

2-/4-fibre MS-SPRing/BLSR protection group

Command syntax: **RTRV-PROTN-GRP**:*tid*:*[aid]*:*ctag*::*protype,[rid]*;

Path protection group

Command syntax: **RTRV-PROTN-GRP**:*tid*:*[aid]*:*ctag*::*protype, , [ppgname][,aid1]*;

The second comma in front of *ppgname* is needed since *rid* is omitted.

The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.

> **Important!** The command **RTRV-PROTN-PPG** which retrieves all used path protection group names is not supported. Therefore its difficult to get an overview about all already used/defined protection group names.
>
> The operator can use the TL1 command **RTRV-PROTN-GRP** and use bay-shelf-#-#-all-all-all as value for the input parameter *aid* in order to derive all path protection groups (see *aid* – Access Identifier in "Command parameters (path protection group)" (p. 3-2762)). Then he gets the used path protection group names per protection group in the command reply.

Input parameters

Command parameters (1+1 Equipment protection group)

1. **`tid`** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **`aid`** – Access Identifier
   Description: The parameter specifies the aid of 1+1 equipment protection group.
   Type: AID
   Range: 1-1-eall or SwitchAndTmgProtGrpAID or CtlProtGroupAID or EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID.

   When using aid 1-1-eall, all actual existing equipment protection groups will be returned:

   - for higher-order switch fabric and timing group (referenced by SwitchAndTmgProtGrpAID)
   - for system controller group (referenced by CtlProtGroupAID)
   - for electrical 155 MBit/s groups (referenced by EP155ProtGroupAID)
   - for electrical 51 MBit/s groups (referenced by EP51ProtGroupAID)
   - for low order switch fabric groups (referenced by LOXCProtGroupAID).
   - for TransMUX groups (referenced by TMUXProtGroupAID)
   - for ES64U groups (referenced by ES64UProtGroupAID)
3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
4. **`protype`** – Protection Type
   The parameter specifies the type of the 1+1 equipment protection group for retrieving the information. The parameter evaluation is case insensitive.
   Type: Alphanumeric value set.

   Range:

   - ALL, if *aid* is 1-1-eall;
   - EQPTSWFBR, if *aid* is SwitchAndTmgProtGrpAID;
   - EQPTCTL, if *aid* is CtlProtGroupAID;
   - EQPTEP155, if *aid* is EP155ProtGroupAID;
   - EQPTEP51, if *aid* is EP51ProtGroupAID;
   - EQPTLOXC, if *aid* is LOXCProtGroupAID.
   - EQPTTMUX, if *aid* is TMUXProtGroupAID.
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

Command parameters (multiplex section/line protection group)

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the *aid* of a 1+1 multiplex section/line protection group or the *aid* of a 1:1 multiplex section/line protection group (only R3.0 and later) or an port *aid* being a member of the group.
   Type: AID
   Range: *bay-shelf*-oall (ranged over OnePlusOne_ProtGrpAID) or OnePlusOne_ProtGrpAID (not ranged) or NByOne_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **protype** – Protection Type
   This parameter specifies the architecture (type) of the protection group. The value for the 1+1 line/MS protection group is 1+1. The value for the 1:1 line/MS protection group is 1xNOPT (only R3.0 and later)
   Type: Value set.
   Range: 1+1 or 1xNOPT

Command parameters (2-/4-fibre MS-SPRing/BLSR protection group)

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID
   Range: *bay-shelf*-tall (ranged over TwoFibre_ProtGrpAID) or bay-shelf-fall (ranged over FourFibre_ProtGrpAID) or TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged).
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: 2F or 4F
5. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: If *aid* is specified, then *rid* can be omitted.

Command parameters (path protection group)

1. **`tid`** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **`aid`** – Access Identifier
   This parameter specifies a path protection group *aid* or an input tributary *aid* being a member of the group in combination with *ppgname*.
   This parameter specifies an input tributary *aid* when used in combination with parameter *aid1*.
   Option: If *ppgname* is specified, then *aid* can be omitted or used as an input tributary aid.
   As *ppgname* is not supported for low order path protections, the aid parameter must specify a path protection group AID for those protection groups.
   The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   If the parameter *aid* specifies a HO tributary on a port of port class EDGE and this tributary is combined with another HO tributary within an OUC, then RTRV-PROTN-GRP shall operate on all protection groups of type PATHDRI identified by these two HO tributaries, if existing.
   This option is also be applied, if wild cards are used in aid. The system discards duplicate responses if several tributaries of the same OUC are within the AID range specified by the wild cards.
   When parameter *aid* is used in combination with parameter *aid1*, then RTRV-PROTN-GRP shall operate on all protection groups of type PATHDRI having the AIDs given in parameters *aid* and *aid1* as input tributaries. In this option the use of wildcards is not allowed.

Type: AID
Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.

3. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the NE in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

4. **`protype`** – Protection Type
   This parameter specifies the protection group type for which the information is retrieved.
   Type: Alphanumeric value set.
   Range: PATHDRI

5. **`ppgname`** – Path protection group Name
   The parameter specifies the path protection group name. Storage and retrieval of the ppgname is case sensitive, but actual use of the ppgname to determine the corresponding protection group is case insensitive.
   Option for high order path protection groups:
   For high order path protections, if aid is specified, then *ppgname* can be omitted.
   For low order path protections, the *ppgname* is omitted as not being supported at all.
   If an input value is specified for *ppgname* and none is specified for *aid*, then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified path protection group name.
   If input values are specified for *aid* and *ppgname* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have the specified *aid*, *ppgname* pair.
   The command will complete successfully if there is no protection group described by *aid*, *ppgname*, but no data will be returned.
   The parameter *ppgname* is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of *ppgname* is not recommended.
   Type: text string
   Range: 1 to 24 characters

6. **`aid1`** – Access Identifier 1
   When parameter *aid1* is applied together with parameter *aid*, these parameters specifies a pair of input tributaries for accessing emulated selector bridges in one command.
   If input values are specified for aid and *aid1* then RTRV-PROTN-GRP operates on all protection groups of type PATHDRI that have these specified tributaries as inputs. If an input value is specified for *aid1*, no value must be specified for

ppgname.
The command will complete successfully if there is no protection group described by aid and *aid1*, but no data will be returned. The parameter ppgname is kept for compatibility reasons only. Because of limitations of the path protection group name aliasing, the use of ppgname is not recommended.
Type: AID
Range: STS1NumAID
The usage of wild cards is not allowed.
The *aid1* parameter is intended to be used in ONNS context. The range is restricted to the tributary types and rates supported by ONNS.

Output format

After receiving the **RTRV-PROTN-GRP** command, the following normal completion response is returned in case of protypes unequal to PATHDRI:

```
   sid date time
M  ctag COMPLD
   ″aid:[spec_block]″
       ...
       ...
   ″aid:[spec_block]″
;
```

After receiving the **RTRV-PROTN-GRP** command, the following normal completion response is returned in case of PATHDRI protection group:

```
   sid date time
M  ctag COMPLD
   ″aid:[spec_block]″
   ″outtribaid,CML:<CML block>″
       .
       .
   ″outtribaid,CML:<CML block>″
       ...
       ...
   ″aid:[spec_block]″
   ″outtribaid,CML:<CML block>″
       .
       .
   ″outtribaid,CML:<CML block>″
;
```

If no protection group entity exists matching the specified values of the input parameters *aid*, *rid* then the following response is returned:

```
   sid date time
M  ctag COMPLD
;
```

If no protection group entity exists matching the specified *ppgname* value, then no response is returned.

Output parameters

Command parameters (1+1 Equipment protection group)

1. **aid** – Access Identifier
   Description: The parameter specifies the aid of 1+1 equipment protection group.
   Type: AID
   Range: SwitchAndTmgProtGrpAID or CtlProtGroupAID or EP155ProtGroupAID or EP51ProtGroupAID or LOXCProtGroupAID or TMUXProtGroupAID or ES64UProtGroupAID.
   All actual existing equipment protection groups will be returned with a specific AID:
   - for higher-order switch fabric and timing group (referenced by SwitchAndTmgProtGrpAID)
   - for system controller group (referenced by CtlProtGroupAID)
   - for electrical 155 MBit/s groups (referenced by EP155ProtGroupAID)
   - for electrical 51 MBit/s groups (referenced by EP51ProtGroupAID)
   - for lower-order switch fabric groups (referenced by LOXCProtGroupAID).
   - for TransMUX groups (referenced by TMUXProtGroupAID)
   - for ES64U groups (referenced by ES64UProtGroupAID)

Command parameters (multiplex section/line protection group)

1. **aid** – Access Identifier
   This parameter specifies the aid of a 1+1 multiplex section/line protection group or the aid of a 1:1 multiplex section/line protection group (only R3.0 and later).
   Type: AID
   Range: OnePlusOne_ProtGrpAID or NByOne_ProtGrpAID

Command parameters (2-fibre MS-SPRing/BLSR protection group)

1. **aid** – Access Identifier
   This parameter specifies the AID of the 2 fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: TwoFibre_ProtGrpAID
   Default: No default

Command parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the 4 fibre MS-SPRing/BLSR protection group.
   Type: AID
   Range: FourFibre_ProtGrpAID

Command parameters (Path protection group)

1. **`aid`** – Access Identifier
   This parameter specifies the AID of the path protection group.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.
2. **`outtribaid`** – Output Tributary Access Identifier
   This parameter specifies the output tributary AID for a constituent signal within the protection group. One value is returned for each constituent signal together with the corresponding CML (Constituent Member List) block.
   Type: AID
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID. The usage of wild cards is not restricted. This range specifies all AIDs which can be used to identify a path protection group. Any defined wild cards (e.g. **`all`** or **`vall`** or **`usall`** in various positions) are allowed.

Spec block parameters (1+1 Equipment protection group)

1. **`actunit`** – Active Unit State
   This parameter specifies which unit is currently selected within the protection group as active unit.
   The value WKG specifies that the worker unit is active.
   The value PROTN specifies that the protection unit is active.
   Type: Alphanumeric value set.
   Range: WKG, PROTN
   Default: For switching protection group instance: The first provisioned/auto-created circuit pack is the default active unit. For controller protection group instance: the active controller unit at the moment of creation of the CTL protection group. For any equipment protection group instance created using the **`ENT-PROTN-GRP`** command: The worker is the default active unit.
2. **`nvmadate`** – NVM Active Date
   This parameter specifies the date of the last successful update of NVM of the current active CTL in the format YY-MM-DD.
   A value of 00-00-00 is reported in case the date information is unavailable.
   Option: This parameter is reported only if *protype==EQPTCTL*, otherwise it is

omitted.
Type: Value set.
Range: YY-MM-DD with value rangesfor YY: 00 to 99for MM: 00 to 12for DD: 00 to 31
Default: None

3. **nvmatime** – NVM Active Time
   This parameter specifies the time of the last successful update of NVM of the current active CTL in the format HH-MM-SS.
   A value of 00-00-00 is reported in case the date information is unavailable.
   Option: This parameter is reported only if *protype==EQPTCTL*, otherwise it is omitted.
   Type: Value set.
   Range: HH-MM-SS with value rangesfor HH: 00 to 23for MM: 00 to 59or SS: 00 to 59
   Default: None
4. **nvmsdate** – NVM Standby Date
   This parameter specifies the date of the last successful update of NVM of the current standby CTL in the format YY-MM-DD.
   A value of 00-00-00 is reported in case the date information is unavailable.
   Option: This parameter is reported only if *protype==EQPTCTL*, otherwise it is omitted.
   Type: Value set.
   Range: YY-MM-DD with value rangesfor YY: 00 to 99for MM: 00 to 12for DD: 00 to 31
   Default: None
5. **nvmstime** – NVM Standby Time
   This parameter specifies the time of the last successful update of NVM of the current standby CTL in the format HH-MM-SS.
   A value of 00-00-00 is reported in case the date information is unavailable.
   Option: This parameter is reported only if *protype==EQPTCTL*, otherwise it is omitted.
   Type: Value set.
   Range: HH-MM-SS with value rangesfor HH: 00 to 23for MM: 00 to 59or SS: 00 to 59
   Default: None
6. **nvmstat** – NVM Synchronization State
   This parameter specifies the current state of NVM synchronisation of the Duplex Control Protection Group.
   Option: This parameter is reported only if *protype==EQPTCTL*, otherwise it is omitted.
   Type: Alphanumeric value set.

Range:

- INSYNC – if NVM of both control units are fully synchronized
- OUTOFSYNC – if either the standby control unit is not accessible or is in the process of re-synchronizing its NVM
- REPAIRING – synchronisation has been interrupted and not yet restarted

Default: None

7. **`protn`** – Protection Unit
   The parameter specifies the protection circuit pack AID being a member of the 1+1 equipment protection group.
   Type: AID
   Range: CpAID, if *protype* is not EQPTLOXC or LoxcAID, if *protype* is EQPTLOXC.
   Default:
   - 1-1-#-#-xcp-cp (if protype is EQPTSWFBR)
   - 1-1-#-#-ctlp-cp (if protype is EQPTCTL)

   For all other equipment protection groups there is no default value as the slot of the protection unit will be dynamically assigned during creation of the group. For rules applying to the selection of the slot for the protection unit refer to ENT-PROTN-GRP.
8. **`protnsiglvl`** – Protection Fault Status
   This parameter specifies the current fault status w.r.t. the protection unit as used by the protection state machine.
   The value EF (Equipment Failed) is used if an equipment defect used as switch criteria is currently present for the protection unit.
   The value NONE is used if no such defect is currently present.
   Type: Alphanumeric value set.
   Range: EF, NONE
   Default: NONE
9. **`protype`** – Protection Type
   This parameter specifies the type of the protection group.
   Type: Alphanumeric value set.
   Range:
   - EQPTSWFBR, if aid is SwitchAndTmgProtGrpAID
   - EQPTCTL, if aid is CtlProtGroupAID
   - EQPTEP155, if aid is EP155ProtGroupAID
   - EQPTEP51, if aid is EP51ProtGroupAID
   - EQPTLOXC, if aid is LOXCProtGroupAID.
   - EQPTTMUX, if aid is TMUXProtGroupAID
   - EQPTES64U, if *aid* is ES64UProtGroupAID.

10. **swreq** – Switch Request State
This parameter specifies the current switch request state of the protection state machine.
The value EQPTFAILED (Equipment Failed) is used if an equipment failed condition is the current state.
The value FS (Forced Switch) is used if a forced switch is the current state.
The value MS (Manual Switch) is used if a manual switch is the current state.
The value NR (No Request) is used if no request is the current state.
Type: Alphanumeric value set.
Range: EQPTFAILED, FS, MS, NR for switch fabric protection and EP155 protection; EQPTFAILED, MS, NR for controller protection.
Default: NR

11. **wkg** – Worker Unit
This parameter specifies the worker circuit pack AID being a member of the 1+1 equipment protection group.
Type: AID
Range: CpAID, if *protype* is not EQPTLOXC or LoxcAID, if *protype* is EQPTLOXC.
Default:
    - 1-1-#-#-xcw-cp (if protype is EQPTSWFBR)
    - 1-1-#-#-ctlw-cp (if protype is EQPTCTL)

For all other equipment protection groups there is no default value as the slot of the worker unit will be dynamically assigned during creation of the group. For rules applying to the selection of the slot for the worker unit refer to ENT-PROTN-GRP.

12. **wkgsiglvl** – Worker Fault Status
This parameter specifies the current fault status w.r.t. worker unit as used by the protection state machine.
The value EF (Equipment Failed) is used if an equipment defect used as switch criteria is currently present for the worker unit.
The value NONE is used if no such defect is currently present.
Type: Alphanumeric value set.
Range: EF, NONE
Default: NONE; i.e. the default is the value ’NONE’

Spec block parameters (multiplex section/line protection group)

1. **`actunit`** – Active Unit State
   This parameter specifies which port is currently selected within the protection group as active port.
   The value WKG specifies that the worker port is active.
   The value PROTN specifies that the protection port is active.
   Type: Alphanumeric value set.
   Range: WKG, PROTN
   Default: WKG
2. **`actunitfe`** – Far-End Incoming Requested Selection
   This parameter is derived from the incoming K1 [5…8] overhead byte.
   The value WKG is used if accepted K byte value is equal to 0.
   The value PRTON is used if the accepted K byte value is equal to 1.
   The value UNAVAILABLE is used if an SF condition is active for the protection line.
   The value NONE is used if there is no request from the far end for 1+1 optimized (only R3.0 and later)
   The value INV (Invalid) is used in all other cases.
   Type: Alphanumeric value set.
   Range: WKG, PROTN, UNAVAILABLE, NONE, INV
   Default: WKG
3. **`actunitne`** – Near-End Outgoing Requested Selection
   This parameter is derived from the sent K1 [5…8] overhead byte.
   The value WKG is used if sent K byte value is equal to 0.
   The value PRTON is used if the sent K byte value is equal to 1.
   Type: Alphanumeric value set.
   Range: WKG, PROTN
   Default: WKG
4. **`aps`** – Far-End Incoming APS
   This parameter is derived from the incoming K1 [1…4] overhead byte.
   It indicates the type of request as follows:
   - DNR: Do Not Revert
   - EXER: Exercise
   - INV: Invalid (unused codes)
   - UNAVAILABLE: Not Available (SF condition pending for protection line)
   - FS: Forced Switch
   - LO: Lockout of Protection
   - MS: Manual Switch
   - NR: No Request
   - RR: Reverse Request
   - SF-H: Signal Fail High

- SF-L: Signal Fail Low
- SD-H: Signal Degrade High
- SD-L: Signal Degrade Low
- WTR: Wait to restore (for MSP 1+1, MSP 1:1 / only used if protection group is revertive) / Wait to rename (for MSP 1+1 Optimized).

Type: Alphanumeric value set.
Range: DNR, EXER, INV, NA, FS, LO , MS, NR, RR, SF-H, SF-L, SD-L, SD-H, WTR
Default: NR

5. **apsout** – Near-End Outgoing APS
This parameter is derived from the sent K1 [1…4] overhead byte.
It indicates the type of request as follows:
- DNR: Do Not Revert
- FS: Forced Switch
- LO: Lockout of Protection
- MS: Manual Switch
- NR: No Request
- RR: Reverse Request
- SF-H: Signal Fail High
- SF-L: Signal Fail Low
- SD-H: Signal Degrade High
- SD-L: Signal Degrade Low
- WTR: Wait to restore (for MSP 1+1, MSP 1:1 / only used if protection group is revertive). Wait to rename (for MSP 1+1 Optimized).

Type: Alphanumeric value set.
Range: DNR, FS, LO , MS, NR, RR, SF-H, SF-L, SD-H, SF-L, WTR
Default: NR

6. **apspn** – MSP-APS ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MSP-APS (Multiplex Section Protection – Automatic protection switching).
Type: Alphanumeric string
Range: 1-24 characters including the character "-".
Default: DEFAULT

7. **`dcclmsprotmd`** – DCC Line/MS Protection Mode
   The parameter specifies, whether the line/MS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set 3;
   Range: ENABLE, DISABLE
   Default: DISABLE
8. **`dccsrsprotmd`** – DCC Sect/RS Protection Mode
   The parameter specifies, whether the section/RS dcc channel is switched as part of the line/MS protection group. The value ENABLE is used, if dcc channel is switched as part of line/MS protection group; otherwise the value DISABLED is used.
   Type: Alphanumeric value set
   Range: ENABLE, DISABLE
   Default: DISABLE
9. **`pgrate`** – protection group Rate
   This parameter is derived from the interface rate of the ports within the protection group. It indicates the same value as for the two ports.
   Type: Alphanumeric value set.
   Range: OC3, OC12, OC48, OC192, OC768, STM1, STM4, STM16, STM64, STM256
   Default: The default value is the interface rate of the worker port.
10. **`protn`** – Protection Unit
   This parameter specifies the protection port AID being a member of the line protection/ MS protection group.
   Type: AID
   Range: SPortAID
11. **`protnsiglvl`** – Protection Fault Status
   This parameter specifies the current fault status w.r.t. port as used by the protection state machine.
   The value SF (Signal Fail) is used if a defect defined as signal fail condition is currently present for the protection port.
   The value SD (Signal Degrade) is used if a defect defined as signal degraded condition is currently present for the protection port.
   The value NONE is used if no such defect is currently present.
   Type: Alphanumeric value set.
   Range: SF, SD, NONE
   Default: NONE
12. **`protocol`** – APS Protocol
   This parameter specifies the protocol used by the protection state machine.
   The value 1+1_UNI is used for 1+1 unidirectional switching.
   The value 1x***N***_BIDIR is used for 1:1 bidirectional switching (***N*** = 1) (only R3.0 and later).
   The value 1+1_OPTM is used for 1+1 optimized, bidirectional switching (only

R3.0 and later). The value 1+1_BIDIR is used for 1+1 bidirectional switching (only R4.0 and later).
The value 1+1_BIDIR is used for 1+1 bidirectional switching (only R4.0 and later).
The value 1+1_UNI shall be used for 1+1 unidirectional switching.
The value 1x***N***_BIDIR shall be used for 1:N bidirectional switching (***N*** = 1..14).
The value 1+1_OPTM shall be used for 1+1 optimized, bidirectional switching.
The value 1+1_BIDIR shall be used for 1+1 bidirectional switching.
Note that “OPTIMIZED” describes rather the protection type than a protocol. Only for compatibility reasons of TL1, the parameter *protocol* is used for its selection.
Type: Alphanumeric value set.
Range: 1+1_UNI, 1xN_BIDIR, 1+1_OPT.
In mind the ***N*** in 1x***N***_BIDIR is considered to be always equal “1”. The value 1+1_OPTM implicitly specifies bidirectional operation mode.
Default: 1+1_UNI, in case of protype = 1+1; 1xN_BIDIR, in case of protype = 1xNOPT (that means protection type 1:1).

13. **`protype`** – Protection Type
This parameter specifies the type of the protection group.
The value for the 1+1 line/MS protection group is “1+1”.
The value for the 1:1 line/MS protection group is “1x***N***OPT” (only R3.0 and later).
In mind the ***N*** in “1x***N***OPT” is considered to be always equal “1”.
Type: Value set.
Range: 1+1 or 1xNOPT.
Default: 1+1

14. **`rme`** – Protection Reversion
The parameter specifies whether revertive or non-revertive mode is used as operation mode of the protection group. The value ENABLE is used if the protection group operates in revertive mode. The value DISABLE is used if the protection group operates in non-revertive mode.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE in case of protype = 1+1; ENABLE in case of protype = 1xNOPT. ENABLE is not valid for protocol = 1+1_OPTM
Default: DISABLE, in case of protype = 1+1; ENABLE, in case of protype = 1xNOPT (i.e. protection type 1:1)

15. **`ssi`** – Secondary Section Indication
The parameter shall indicate the ’Secondary Section’ (line).
Type: AID
Range: WKG, PROTN; WKG denotes working section 1, PROTN denotes working section 2 according ITU-T G.841 Annex B.
Default: PROTN

16. **`swreq`** – Switch Request State
This parameter specifies the current switch request state of the protection state machine.

It shall have the following values:

- DNR: Do Not Revert
- FS: Forced Switch
- LO: Lockout of Protection
- MS: Manual Switch
- SF: Signal Fail
- SD: Signal Degrade
- WTR: Wait to restore (for MSP 1+1, MSP 1:1 / only used if protection group is revertive). Wait to rename (for MSP 1+1 Optimized)
- FOP: Failure Of Protocol
- NR: No Request.

Type: Alphanumeric value set.
Range: DNR, FS, LO , MS, SF, SD, WTR, FOP, NR
Default: NR

17. **`wkg`** – Worker Unit
This parameter specifies the worker port AID being a member of the 1+1 line/MS protection group.
Type: AID
Range: SPortAID

18. **`wkgsiglvl`** – Worker Fault Status
This parameter specifies the current fault status w.r.t. port as used by the protection state machine.
The value SF (Signal Fail) is used if a defect defined as signal fail condition is currently present for the worker port.
The value SD (Signal Degrade) is used if a defect defined as signal degraded condition is currently present for the worker port.
The value NONE issued if no such defect is currently present.
Type: Alphanumeric value set.
Range: SF, SD, NONE.
Default: NONE.

19. **`wtr`** – Wait To Restore Time / Wait to Rename Time
The parameter specifies the “wait to restore” time used by the protection state machine operating in revertive mode. In case of MSP 1+1 optimized, the same parameter is used for the “wait to rename” time.
Type: numeric string [minute]
Range: 0 – 60
Default: 5.

Spec block parameters (2-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
   If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value manually with ED-MAP-RING and supply the ring map information to each node on the ring for the BLSR protocol to operate. It is necessary to activate this user-provisioned information with SET-MAP-RING. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
   Type: Alphanumeric string.
   Range: ENABLE, DISABLE.
   Default: ENABLE.
2. **`asmc`** – Automatic Squelch Map Calculation
   If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
   Type: Alphanumeric string.
   Range: ENABLE, DISABLE (default).
3. **`blsrpn`** – MS-SPRing-BLSR ASAP Pointer
   This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
   Type: Alphanumeric string
   Range: 1-24 characters including the character “-”.
   Default: DEFAULT
4. **`eapschst`** – Incoming East APS Channel Status
   This parameter is derived from the incoming K2 [6..8] overhead byte of the east port.

   It indicates the status as follows:

   - IDLE: Idle
   - BRIDGED: Bridged
   - BANDS: Bridged & Switched
   - XTRFC: Extra Traffic on protection channels
   - NOTUSED: Not Used – Reserved for future use
   - RDIL: RDI-L/MS-RDI (Remote Defect Indication -Line/Multiplex Section)
   - AISL: AIS-L/MS-AIS (Alarm Indication Signal – Line/Multiplex Section)

   Type: Alphanumeric value set.
   Range: IDLE, BRIDGED, BANDS, XTRFC, NOTUSED, RDIL, AISL
   Default: IDLE

5. **`eapsdnid`** – Incoming East APS Destination Node Id
   This parameter is derived from the incoming K1 [5..8] overhead byte of the east port. It specifies the node ID for which the K1 byte is destined.
   The value UNAVAILABLE is used in case of an pending SF condition.
   Type: Alphanumeric value set.
   Range: 0-15 or UNAVAILABLE
   Default: 0
6. **`eapsprotn`** – Incoming East APS Protection Path
   This parameter is derived from the incoming K2 overhead byte of the east port.
   It indicates the protection path as follows:
   - 0: short path code (S)
   - 1: long path code (L)

   The value UNAVAILABLE is used in case of an pending SF condition.
   Type: Alphanumeric value set.
   Range: 0, 1, UNAVAILABLE
   Default: 0
7. **`eapssnid`** – Incoming East APS Source Node Id
   This parameter is derived from the incoming K2 [1..4] overhead byte of the east port. It specifies the node ID which sourced the K1 byte.
   The value UNAVAILABLE is used in case of an pending SF condition.
   Range: 0-15 or UNAVAILABLE
   Default: 0
8. **`eapssw`** – Incoming East APS Switch Priority
   This parameter is derived from the incoming K1 [1..4] overhead byte of the east port.
   It indicates the switch priority/bridge request code as follows:
   - LPS: Lockout of Protection Span
   - FSS: Forced Switch Span (4F only)
   - FSR: Forced Switch Ring
   - SFS: Signal Fail Span (4F only)
   - SFR: Signal Fail Ring
   - SDP: Signal Degrade Protection (4F only)
   - SDS: Signal Degrade Span (4F only)
   - SDR: Signal Degrade Ring
   - MSS: Manual Switch Span (4F only)
   - MSR: Manual Switch Ring
   - WTR: Wait to Restore
   - EXERS: Exerciser Span (4F only)
   - EXERR: Exerciser Ring
   - RRS: Reverse Request Span

- RRR: Reverse Request Ring
- NR: No Request.

The value UNAVAILABLE is used in case of an pending SF condition.
Range: LPS, FSS, FSR, SFS, SFR, SDP, SDS, SDR, MSS, MSR, WTR, EXERS, EXERR, RRS, RRR, NR, UNAVAILABLE.
Default: NR.

9. **`east`** – East Port
This parameter specifies the port being a member in the role of the east port for a 2-fibre MS-SPRing/BLSR protection group.
Type: AID
Range: SPortAID

10. **`elprotnsw`** – East Last protection switch Attempt
This parameter specifies, whether the last switch request for the east side succeeded or not.
The values have the following semantics:
    - SUCCESS: last request did succeed
    - DENY: last request was denied due a validation rule
    - FAILURE: last request could not be completed because far end node did not response with proper K-byte.

    Type: Alphanumeric value set.
Range: SUCCESS, DENY, FAILURE
Default: SUCCESS.

11. **`eswreq`** – East Switch Request State
This parameter specifies the active switch request state of the protection state machine for the east side. If the east side is an associated side of a switching node, the value will be derived from the request currently associated with the concerned side, and accepted in the protocol as highest or coexisting priority. For all other cases, the value is 'NR'.
It has the following values:
    - LPS: Lockout of Protection Span
    - FSR: Forced Switch of Service to Protection – Ring
    - SFR: Signal Fail – Ring
    - SDR: Signal Degrade – Ring
    - MSR: Manual Switch of Service to Protection – Ring
    - WTR: Wait to Restore
    - NR: No Request.

    Type: Alphanumeric value set.
Range LPS, FSR, SFR, SDR, MSR, WTR, NR.
Default: NR.

12. **`pgrate`** – protection group Rate
    This parameter is derived from the interface rate of the ports within the protection group. It indicates the same value as for the two ports.
    Type: Alphanumeric value set.
    Range: OC48, OC192, OC768, STM16, STM64, STM256
    Default: The default value is the interface rate of the east port.
13. **`protype`** – Protection Type
    This parameter specifies the type of the protection group. The value for the 2-fibre MS-SPRing/BLSR protection group is 2F.
    Type: Alphanumeric value set.
    Range: 2F.
14. **`rcamode`** – Ring Circuit Alarm Mode
    This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If *rcamode* is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
    Type: Alphanumeric value set.
    Range: ENABLE (default), DISABLE
15. **`rid`** – Ring Identification
    This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
    Type: text string of chars allowed for TID strings.
    Range: 1 to 15 characters.
    Default: NONE.
16. **`ringndst`** – Ring Node APS State
    This parameter specifies the ring node APS state.
    It shall have the following values under normal conditions: IDLE, SWITCHING, PASSTHROUGH
    It shall have in addition the following values if protection state machine is suspended: SUSPENDED-IDLE, SUSPENDED-SWITCHING, SUSPENDED-PASSTHROUGH
    Type: Alphanumeric value set.
    Range: IDLE, SWITCHING, PASSTHROUGH, SUSPENDED-IDLE, SUSPENDED-SWITCHING, SUSPENDED-PASSTHROUGH.
    Default: SUSPENDED-IDLE.
17. **`wapschst`** – Incoming West APS Channel Status
    This parameter is derived from the incoming K2 [6..8] overhead byte of the west port.

It indicates the status as follows:

- IDLE: Idle BRIDGED Bridged
- BANDS: Bridged & Switched
- XTRFC: Extra Traffic on protection channels
- NOTUSED: Not Used – Reserved for future use
- RDIL: RDI-L/MS-RDI (Remote Defect Indication – Line/Multiplex Section).
- AISL: AIS-L/MS-AIS (Alarm Indication Signal – Line/Multiplex Section).

Type: Alphanumeric value set.
Range: IDLE, BRIDGED, BANDS, XTRFC, NOTUSED, RDIL, AISL.
Default: IDLE.

18. **`wapsdnid`** – Incoming West APS Destination Node Id
This parameter is derived from the incoming K1 [5..8] overhead byte of the west port. It specifies the node ID for which the K1 byte is destined. The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric string.
Range: 0-15 or UNAVAILABLE.
Default: 0.

19. **`wapsprotn`** – Incoming West APS Protection Path
This parameter is derived from the incoming K2 overhead byte of the west port.
It indicates the protection path as follows:
   - 0: short path code (S)
   - 1: long path code (L)

   The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric value set.
Range: 0, 1, UNAVAILABLE.
Default: 0.

20. **`wapssnid`** – Incoming West APS Source Node Id
This parameter is derived from the incoming K2 [1..4] overhead byte of the west port. It specifies the node ID which sourced the K1 byte. The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric string.
Range: 0-15 or UNAVAILABLE.
Default: 0.

21. **`wapssw`** – Incoming West APS Switch Priority
This parameter is derived from the incoming K1 [1..4] overhead byte of the west port.
It indicates the switch priority/bridge request code as follows:
   - LPS: Lockout of Protection Span
   - FSS: Forced Switch Span (4F only)
   - FSR: Forced Switch Ring
   - SFS: Signal Fail Span (4F only)

- SFR: Signal Fail Ring
- SDP: Signal Degrade Protection (4F only)
- SDS: Signal Degrade Span (4F only)
- SDR: Signal Degrade Ring
- MSS: Manual Switch Span (4F only)
- MSR: Manual Switch Ring
- WTR: Wait to Restore
- EXERS: Exerciser Span (4F only)
- EXERR: Exerciser Ring
- RRS: Reverse Request Span
- RRR: Reverse Request Ring
- NR: No Request.

The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric value set.
Range: LPS, FSS, FSR, SFS, SFR, SDP, SDS, SDR, MSS, MSR, WTR, EXERS, EXERR, RRS, RRR, NR, UNAVAILABLE.
Default: NR.

22. **`west`** – West Port
This parameter specifies the port being a member in the role of the west port for a 2-fibre MS-SPRing/BLSR protection group.
Type: AID.
Range: SPortAID.
Default: None.

23. **`wlprotnsw`** – West Last protection switch Attempt
This parameter specifies, whether the last switch request for the west side succeeded or not.

The values have the following semantics:

- SUCCESS: last request did succeed
- DENY: last request was denied due a validation rule
- FAILURE: last request could not be completed because far end node did not response with proper K-byte.

Type: Alphanumeric value set.
Range: SUCCESS, DENY, FAILURE
Default: SUCCESS.

24. **`wswreq`** – West Switch Request State
This parameter specifies the active switch request state of the protection state machine for the east side. If the east side is an associated side of a switching node, the value is derived from the request currently associated with the concerned side, and accepted in the protocol as highest or coexisting priority. For all other cases, the value is ’NR’.

It have the following values:

- LPS: Lockout of Protection Span
- FSR: Forced Switch of Service to Protection – Ring
- SFR: Signal Fail – Ring
- SDR: Signal Degrade – Ring
- MSR: Manual Switch of Service to Protection – Ring
- WTR: Wait to Restore
- NR: No Request.

Type: Alphanumeric value set.
Range LPS, FSR, SFR, SDR, MSR, WTR, NR.
Default: NR.

25. **`wtr`** – Wait To Restore Time
This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

Spec block parameters (4-fibre MS-SPRing/BLSR protection group)

1. **`ard`** – Automatic Ring Discovery
If this parameter is enabled, the automatic ring discovery algorithm will assign nodeID values to each node in the ring and construct the necessary ring map information. All nodes in the ring must support the Ring Discovery interworking. If disabled, the user will assign the NodeID value and supply the ring map information to each node on the ring for the BLSR protocol to operate. Until all nodes on the ring have the same ring map, nodes will have most likely an outstanding NodeID mismatch alarm active.
Type: Alphanumeric string.
Range: ENABLE, DISABLE.
Default: ENABLE.

2. **`asmc`** – Automatic Squelch Map Calculation
If this parameter is enabled, the automatic squelch map calculation is switched on. If disabled, the function is switched off.
Type: Alphanumeric string.
Range: ENABLE, DISABLE (default).

3. **blsrpn** – MS-SPRing-BLSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type MS-SPRing-BLSR.
Type: Alphanumeric string.
Range: 1-24 characters including the character “-”.
Default: DEFAULT.
4. **eactunit** – East Active Unit State
The parameter specifies whether the traffic associated with the east worker port is physically running over the east worker port (i.e. value WKG is used in that case) or over a protection port (i.e. value PROTN is used in that case). The protection port can be the east protection port in case of span switch or the west protection port in case of a loopback switch. The value of the parameter is derived from the internal handled active line state of the service port and the active line state of the protection ports. *eactunit* = WKG: active line state of east worker port is active. *eactunit* = PROTN: active line state of east worker port is standby.
Type: Alphanumeric value set.
Range: WKG, PROTN.
Default: WKG.
5. **eapschst** – Incoming East APS Channel Status
This parameter is derived from the incoming K2 [6..8] overhead byte of the east port.
It indicates the status as follows:
   - IDLE: Idle
   - BRIDGED: Bridged
   - BANDS: Bridged & Switched
   - XTRFC: Extra Traffic on protection channels
   - NOTUSED: Not Used – Reserved for future use
   - RDIL: RDI-L/MS-RDI (Remote Defect Indication -Line/MultiplexSection)
   - AISL: AIS-L/MS-AIS (Alarm Indication Signal – Line/Multiplex Section)
   - Extension for TOP-EX
     In case of *protnmode* = TOPEX the following additional value is supported: (ACT) command to ACT to complete the Switch.

   Type: Alphanumeric value set.
Range: IDLE, BRIDGED, BANDS, XTRFC, NOTUSED, ACT, RDIL, AISL.
Default: IDLE.
6. **eapsdnid** – Incoming East APS Destination Node Id
This parameter is derived from the incoming K1 [5..8] overhead byte of the east port. It specifies the node ID for which the K1 byte is destined.
The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric string.
Range: 0-15 or UNAVAILABLE.
Default: 0.

7. **`eapsprotn`** – Incoming East APS Protection Path
   This parameter is derived from the incoming K2 overhead byte of the east port.
   It indicates the protection path as follows:
   - 0: short path code (S)
   - 1: long path code (L)

   The value UNAVAILABLE is used in case of an pending SF condition.
   Type: Alphanumeric value set.
   Range: 0, 1, UNAVAILABLE.
   Default: 0.
8. **`eapssnid`** – Incoming East APS Source Node Id
   This parameter is derived from the incoming K2 [1..4] overhead byte of the east port. It specifies the node ID which sourced the K1 byte.
   The value UNAVAILABLE is used in case of an pending SF condition.
   Type: Alphanumeric string.
   Range: 0-15 or UNAVAILABLE.
   Default: 0.
9. **`eapssw`** – Incoming East APS Switch Priority
   This parameter is derived from the incoming K1 [1..4] overhead byte of the east port.
   It indicates the switch priority/bridge request code as follows:
   - LPS: Lockout of Protection Span
   - FSS: Forced Switch Span (4F only)
   - FSR: Forced Switch Ring
   - SFS: Signal Fail Span (4F only)
   - SFR: Signal Fail Ring
   - SDP: Signal Degrade Protection (4F only)
   - SDS: Signal Degrade Span (4F only)
   - SDR: Signal Degrade Ring
   - MSS: Manual Switch Span (4F only)
   - MSR: Manual Switch Ring
   - WTR: Wait to Restore
   - EXERS: Exerciser Span (4F only)
   - EXERR: Exerciser Ring
   - RRS: Reverse Request Span

- RRR: Reverse Request Ring
- NR: No Request.

The value UNAVAILABLE is used in case of an pending SF condition.
The bridge request LPS and SFP (Signal Fail Protection) share a common code value. Therefore they cannot be distinguished. The value 'LPS' will be used in that context.
Type: Alphanumeric value set.
Range: LPS, FSS, FSR, SFS, SFR, SDP, SDS, SDR, MSS, MSR, WTR, EXERS, EXERR, RRS, RRR, NR, UNAVAILABLE.
Default: NR.

10. **`eastprotn`** – East Protection Port
This parameter shall specifies the port being a member in the role of the east protection port for a 4-fibre MS-SPRing/BLSR protection group.
Type: AID
Range: SPortAID

11. **`eastwkg`** – East Worker Port
This parameter specifies the port being a member in the role of the east worker port for a 4-fibre MS-SPRing/BLSR protection group.
Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID.
Range: In case of an asymmetric ring: NOTUSED. In all other cases: SPortAID.

12. **`elprotnsw`** – East Last protection switch Attempt
This parameter specifies, whether the last switch request for the east side succeeded or not.

The values have the following semantics:

- SUCCESS: last request did succeed
- DENY: last request was denied due a validation rule
- FAILURE: last request could not be completed because far end node did not response with proper K-byte.

Type: Alphanumeric value set.
Range: SUCCESS, DENY, FAILURE.
Default: SUCCESS.

13. **`eswreq`** – East Switch Request State
This parameter specifies the active switch request state of the protection state machine for the east side. If the east side is an associated side of a switching node, the value is derived from the request currently associated with the concerned side, and accepted in the protocol as highest or coexisting priority. For all other cases, the value is 'NR'.

It has the following values:

- LPS: Lockout of Protection Span
- FSS: Forced Switch of Service to Protection – Span
- FSR: Forced Switch of Service to Protection – Ring
- SFS: Signal Fail – Span

- SFR: Signal Fail – Ring
- SDS: Signal Degrade – Span
- SDR: Signal Degrade – Ring
- SDP : Signal Degrade – Protection
- MSS: Manual Switch of Service to Protection – Span
- MSR: Manual Switch of Service to Protection – Ring
- WTR: Wait to Restore
- NR: No Request.

Type: Alphanumeric value set.
Range: LPS, FSS, FSR, SFS, SFR, SDS, SDR, SDP, MSS, MSR, WTR, NR The bridge request LPS and SFP (Signal Fail Protection) share a common code value. Therefore they cannot be distinguished. The value 'LPS' will be used in that context.
Default: NR.

14. **`pgrate`** – protection group rate
This parameter is derived from the interface rate of the ports within the protection group. It indicates the same value as for the four ports.
Type: Alphanumeric value set.
Range: OC48, OC192, STM16, STM64.
If *protnmode* = TOPEX, then the range is restricted to: STM64 only
Default: The default value is the interface rate of the east port.

15. **`protnmode`** – Protection Switch Mode
The parameter specifies the protection switch mode. The value LOOPBACK is used for the classical ring application. The value SHORTENED is used for transmission delay sensitive applications. The value TOPEX (only R3.0 – R3.2 and R8.0.1 and later) is used for the so called transoceanic application, which allows extra traffic, improves restorability and improves delay as well as in SHORTENED mode.
In case of a ring switch … LOOPBACK performs the bridge and switch function in that nodes which are most close to the failed span. SHORTENED performs the bridge function at the add node repectively the switch function at the drop node of each ring circuit. It is an application only similar to the transoceanic protocol.
TOPEX allows, compared to SHORTENED mode, additionally the traffic restoration of not affected spans. It is the transoceanic protocol application.
Consistent Protection Switch Mode Setting
The Protection Switch Mode must be set consistently on all nodes in the ring. E.g. for all UNITE NEs in a TOP+EX ring the parameter protnmode must be set to TOPEX. In a ring with inconsistent Protection Switch Mode, there will be no protection switching. The UNITE NEs will raise an Inconsistent Ring Protection Mode alarm.
Type: Alphanumeric value set.
Range: LOOPBACK, SHORTENED, TOPEX.
Default: LOOPBACK.

16. **`protype`** – Protection Type
This parameter specifies the type of the protection group. The value for the 4-fibre MS-SPRing/BLSR protection group is 4F.
Type: Alphanumeric value set.
Range: 4F.

17. **`rcamode`** – Ring Circuit Alarm Mode
This parameter can be used to enable or disable ring circuit alarms for the local MS SPRING / BLSR protection group. Ring Circuit alarms are results of the ring circuit validation (Path Integrity Failure) and of the audit of the local squelch map (Local SQuelch Map Conflict). The parameter does not control the validation or audit itself, but only the associated alarming. If *rcamode* is DISABLED, any alarm associated with the local cross-connections will be suppressed and not issued.
Type: Alphanumeric value set.
Range: ENABLE (default), DISABLE

18. **`rid`** – Ring Identification
This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
Type: text string of chars allowed for TID strings.
Range: 1 to 15 characters.
Default: NONE

19. **`ringndst`** – Ring Node APS State
This parameter specifies the ring node APS state.
It shall have the following values under normal conditions: IDLE, SWITCHING, PASSTHROUGH
It shall have in addition the following values if protection state machine is suspended: SUSPENDED-IDLE, SUSPENDED-SWITCHING, SUSPENDED-PASSTHROUGH.
Type: Alphanumeric value set.
Range: IDLE, SWITCHING, PASSTHROUGH, SUSPENDED-IDLE, SUSPENDED-SWITCHING, SUSPENDED-PASSTHROUGH.
Default: SUSPENDED-IDLE.

20. **`swtr`** – Span Wait To Restore Time
The parameter specifies the wait to restore time for span switches used by the protection state machine operating in revertive mode. The protection shall operate in revertive mode if the wait-to-restore time is in the range 0-12. The protection shall operate in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.

Note: If *wtr* time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

21. **`wactunit`** – West Active Unit State
The parameter specifies whether the traffic associated with the west worker port is physically running over the west worker port (i.e. value WKG is used in that case) or over a protection port (i.e. value PROTN is used in that case). The protection port can be the west protection port in case of span switch or the east protection port in case of a loopback switch. The value of the parameter is derived from the internal handled active line state of the service port and the active line state of the protection ports. wactunit = WKG: active line state of west worker port is active wactunit = PROTN: active line state of west worker port is standby.
Type: Alphanumeric value set.
Range: WKG, PROTN.
Default: WKG.

22. **`wapschst`** – Incoming West APS Channel Status
This parameter is derived from the incoming K2 [6..8] overhead byte of the west port.
It indicates the status as follows:
    - IDLE: Idle
    - BRIDGED: Bridged
    - BANDS: Bridged & Switched
    - XTRFC: Extra Traffic on protection channels
    - NOTUSED: Not Used – Reserved for future use
    - RDIL: RDI-L/MS-RDI (Remote Defect Indication – Line/Multiplex Section)
    - AISL: AIS-L/MS-AIS (Alarm Indication Signal – Line/Multiplex Section)
    - Extension for TOP-EX
    In case of *protnmode* = TOPEX the following additional value is supported: (ACT) command to ACT to complete the Switch.

    Type: Alphanumeric value set.
    Range: IDLE, BRIDGED, BANDS, XTRFC, NOTUSED, ACT, RDIL, AISL.
    Default: IDLE.

23. **`wapsdnid`** – Incoming West APS Destination Node Id
This parameter is derived from the incoming K1 [5..8] overhead byte of the west port. It specifies the node ID for which the K1 byte is destined. The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric string.
Range: 0-15 or UNAVAILABLE.
Default: 0.

24. **wapsprotn** – Incoming West APS Protection Path
This parameter is derived from the incoming K2 overhead byte of the west port.
It indicates the protection path as follows:
- 0: short path code (S)
- 1: long path code (L)

The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric value set.
Range: 0, 1, UNAVAILABLE.
Default: 0.

25. **wapssnid** – Incoming West APS Source Node Id
This parameter is derived from the incoming K2 [1..4] overhead byte of the west port. It specifies the node ID which sourced the K1 byte. The value UNAVAILABLE is used in case of an pending SF condition.
Type: Alphanumeric string.
Range: 0-15 or UNAVAILABLE.
Default: 0.

26. **wapssw** – Incoming West APS Switch Priority
This parameter is derived from the incoming K1 [1..4] overhead byte of the west port.
It indicates the switch priority/bridge request code as follows:
- LPS: Lockout of Protection Span
- FSS: Forced Switch Span
- FSR: Forced Switch Ring
- SFS: Signal Fail Span
- SFR: Signal Fail Ring
- SDP: Signal Degrade Protection
- SDS: Signal Degrade Span
- SDR: Signal Degrade Ring
- MSS: Manual Switch Span
- MSR: Manual Switch Ring
- WTR: Wait to Restore
- EXERS: Exerciser Span
- EXERR: Exerciser Ring
- RRS: Reverse Request Span

- RRR: Reverse Request Ring
- NR: No Request

The value UNAVAILABLE is used in case of an pending SF condition.
The bridge request LPS and SFP (Signal Fail Protection) share a common code value. Therefore they cannot be distinguished. The value 'LPS' will be used in that context.
Type: Alphanumeric value set.
Range: LPS, FSS, FSR, SFS, SFR, SDP, SDS, SDR, MSS, MSR, WTR, EXERS, EXERR, RRS, RRR, NR, UNAVAILABLE.
Default: NR.

27. **`westprotn`** – West Protection Port
This parameter specifies the port being a member in the role of the west protection port for a 4-fibre MS-SPRing/BLSR protection group.
Type: AID
Range: SPortAID

28. **`westwkg`** – West Worker Port
This parameter specifies the port being a member in the role of the west worker port for a 4-fibre MS-SPRing/BLSR protection group
Type: In case of an asymmetric ring: alphanumeric string; in all other cases: AID
Range: In case of an asymmetric ring: NOTUSED; in all other cases: SPortAID.

29. **`wlprotnsw`** – West Last protection switch Attempt
This parameter specifies, whether the last switch request for the west side succeeded or not.

The values have the following semantics:

- SUCCESS: last request did succeed
- DENY: last request was denied due a validation rule
- FAILURE: last request could not be completed because far end node did not response with proper K-byte

Type: Alphanumeric value set.
Range: SUCCESS, DENY, FAILURE.
Default: SUCCESS.

30. **`wswreq`** – West Switch Request State
This parameter specifies the active switch request state of the protection state machine for the east side. If the east side is an associated side of a switching node, the value is derived from the request currently associated with the concerned side, and accepted in the protocol as highest or coexisting priority. For all other cases, the value is 'NR'.

It have the following values:

- LPS: Lockout of Protection Span
- FSS: Forced Switch of Service to Protection – Span
- FSR: Forced Switch of Service to Protection – Ring
- SFS: Signal Fail – Span

- SFR: Signal Fail – Ring
- SDS: Signal Degrade – Span
- SDR: Signal Degrade – Ring
- SDP: Signal Degrade – Protection
- MSS: Manual Switch of Service to Protection – Span
- MSR: Manual Switch of Service to Protection – Ring
- WTR: Wait to Restore
- NR: No Request.

Type: Alphanumeric value set.
Range LPS, FSS, FSR, SFS, SFR, SDS, SDR, SDP, MSS, MSR, WTR, NR The bridge request LPS and SFP (Signal Fail Protection) share a common code value. Therefore they cannot be distinguished. The value 'LPS' will be used in that context.
Default: NR.

31. **`wtr`** – Wait To Restore Time
This parameter specifies the wait to restore time for ring switches used by the protection state machine operating in revertive mode. The protection operates in revertive mode if the wait-to-restore time is in the range 0-12. The protection operates in non-revertive mode if the wait-to-restore time has the value 99 representing the infinite value.
Note: If wtr time is set to infinite, the switch request state is WTR until a higher request occurs.
Type: numeric string [minute].
Range: 0 – 12 in 1 minute increments and 99.
Default: 5.

Spec block parameters (path protection group)

1. **`cst`** – Constituent Signal Type
This parameter specifies the rate of the path protection group.
Type: Alphanumeric value set.
Range: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C
2. **`dripathpn`** – SNCP-UPSR ASAP Pointer
This parameter specifies the value for the ASAP (alarm severity assignment profile) name of type SNCP-UPSR.
Type: Alphanumeric string.
Range: 1-24 characters including the character "-".
Default: DEFAULT.

3. **`htmr`** – Hold-Off Time
   This parameter specifies the hold-off time for automatic protection switching used by the protection state machine.
   Type: numeric string [ms].
   Range for high order path protection groups: 0 .. 99 ms in 1 ms steps and 100 .. 10000 ms in 100 ms steps.
   Range for low order path protection groups: 0, 25, 50, 75 ms (in steps of 25 ms) and 100 .. 10000 ms in 100 ms steps.
4. **`ppgname`** – Path protection group Name
   This parameter specifies the path protection group name. Storage and retrieval of the *ppgname* is case sensitive, but actual use of the *ppgname* to determine the corresponding protection group is case insensitive.
   This parameter is omitted for low order path protection groups.
   Type: text string.
   Range: 0 to 24 characters.
   Default: empty string
5. **`protn`** – Protection Tributary
   This parameter specifies the protection input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
   The usage of wild cards is not allowed for any part of the AID.
6. **`protype`** – Protection Type
   This parameter specifies the type of the protection group. The value for the path protection group is PATHDRI.
   Type: Alphanumeric value set.
   Range: PATHDRI.
   Default: PATHDRI.
7. **`rme`** – Protection Reversion
   This parameter specifies, whether revertive or non-revertive mode is used as operation mode of the protection group.
   The value ENABLE is used if the protection operates in revertive mode.
   The value DISABLE is used if the protection operates in non-revertive mode.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
8. **`snctype`** – SNC Protection Type
   This parameter specifies the type of the path protection w.r.t. switch criteria used by the protection state machine.
   The value SNCI (Inherent) is used if server signal fail is used as switch criteria.
   The value SNCN (Non-Instrusive) is used if server signal fail and defects

supervised by a non-intrusive path monitor are used as switch criteria
The value NA is used if one of the tributary inputs is related to a port operating in SONET mode.
Type: Alphanumeric value set.
Range: SNCI, SNCN, NA.

9. **`wkg`** – Worker Tributary
   This parameter specifies the worker input tributary AID being a member of the path protection group.
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID or TMuxAPTNumAID.
   The usage of wild cards is not allowed for any part of the AID.

10. **`wtr`** – Wait To Restore Time
    This parameter specifies the wait to restore time used by the protection state machine operating in revertive mode.
    Type: numeric string [minute].
    Range: 0 – 60.

CML Block parameters (path protection group)

The CML (Constituent Member List) block is only used in the context of the path protection group. One block will be returned for each constituent signal within the protection group.

1. **`actunit`** – Active Unit State
   This parameter specifies which input tributary is currently selected within the protection group as active input.
   The value WKG specifies that the worker tributary is active.
   The value PROTN specifies that the protection tributary is active.
   Type: Alphanumeric value set.
   Range: WKG, PROTN.
   Default: WKG.

2. **`actunitaid`** – Active Unit State AID
   This parameter specifies the input tributary AID of the constituent signal which is currently selected within the protection group as active input for the dedicated constituent tributary output (see parameter outtribaid).
   Type: AID.
   Range: STS1NumAID or VCGNumAID or GenLOTribAID.
   The usage of wild cards is not allowed for any part of the AID.
   Default: AID of the worker tributary.

3. **`cst`** – Constituent Signal Type
   This parameter specifies the rate of the constituent signal within the path protection group.
   Type: Alphanumeric value set.
   Range: LOVT1, STS1, STS3C, STS12C, STS48C, STS192C, LOVC12, LOVC3, VC3, VC4, VC44C, VC416C, VC464C.
   Default: The default value is derived from the tributary input signal rate list in case of fixed mode.
4. **`protnsiglvl`** – Protection Fault Status
   This parameter specifies the current fault status w.r.t. tributary as used by the protection state machine.
   The value SF (Signal Fail) is used if a defect defined as signal fail condition is currently present for the protection tributary.
   The value SD (Signal Degrade) is used if a defect defined as signal degrade condition is currently present for the protection tributary.
   The value NONE is used if no such defect is currently present.
   Type: Alphanumeric value set.
   Range: SF, SD, NONE.
   Default: NONE.
5. **`swreq`** – Switch Request State
   This parameter specifies the current switch request state of the protection state machines.

   It have the following values:

   - LO: Lockout of Protection
   - FS: Forced Switch
   - MS: Manual Switch
   - SF: Signal Fail
   - SD: Signal Degrade
   - WTR: Wait to restore (only used if protection group is revertive)
   - NR: No Request.

   Type: Alphanumeric value set.
   Range: LO ,FS, MS, SF, SD , WTR, NR.
   Default: NR.
6. **`wkgsiglvl`** – Worker Fault Status
   This parameter specifies the current fault status w.r.t. tributary as used by the protection state machine.
   The value SF (Signal Fail) is used if a defect defined as signal fail condition is currently present for the worker tributary.
   The value SD (Signal Degrade) is used if a defect defined as signal degrade

condition is currently present for the worker tributary.
The value NONE is used if no such defect is currently present.
Type: Alphanumeric value set.
Range: SF, SD, NONE.
Default: NONE.

Detailed behavior description

The system shall allow to retrieve path protection groups at VCG tributaries using the AID format of an AU-3/STS-1 Tributary. I.e. the VCG number specified in the port field of the AID is allowed with or without the leading ″v″.

Example input/output

1+1 Equipment Protection Group

The following example retrieves 1+1 equipment protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-estgrp:123456::EQPTSWFBR;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-estgrp:protype=EQPTSWFBR,wkg=1-1-#-#-xcw-cp,protn=1-1-#-#-xcp-cp,actunit=
WKG,protnsiglvl=NONE,wkgsiglvl=NONE,swreq=NR″
;
```

The following example retrieves 1+1 equipment protection group information for the Duplex Control Protection Group while both NVMs are fully synchronized:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-ectlgrp:123456::EQPTCTL;
   LT-UNITE 06-10-11 11:15:00
M  123456 COMPLD
   ″1-1-ectlgrp:protype=EQPTCTL,actunit=WKG,nvmadate=06-10-11,nvmatime=09-17-
      43,nvmsdate=06-10-11,nvmstime=09-17-43,nvmstat=INSYNC,protn=1-1-#-#-ctlp-
      cp,protnsiglvl=NONE,swreq=NR,wkg=1-1-#-#-ctlw-cp,wkgsiglvl=NONE″
;
```

The following example retrieves TransMUX equipment protection group information:

```
RTRV-PROTN-GRP:UNITE-NE:1-1-etmux01:123456::EQPTTMUX;
   UNITE-NE 07-12-21 09:00:00
M  123456 COMPLD
   ″1-1-etmux01:protype=EQPTTMUX,wkg=1-1-#-#-21-cp,protn=1-1-#-#-22-cp,actunit=
WKG,protnsiglvl=NONE,wkgsiglvl=NONE,swreq=NR″
;
```

Multiplex section/line protection group (1+1)

The following example retrieves 1+1 multiplex section/line protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-o001:123456::1+1;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-o01:protype=1+1,wkg=1-1-#-#-13-1,protn=1-1-#-#-14-1,wtr=5,apspn=DEFAULT,
      dcclmsprotmd=DISABLE,dccsrsprotmd=DISABLE,protocol=1+1_UNI,rme=
      DISABLE,swreq=NR,actunit=WKG,actunitfe=WKG,
      actunitne=WKG,aps=NR,apsout=NR,pgrate=OC192,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

The following example retrieves 1+1 optimized multiplex section/line protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-o001:123456::1+1;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-o001:protype=1+1,wkg=1-1-#-#-13-1,protn=1-1-#-#-14-1,wtr=5,apspn=DEFAULT,
      dcclmsprotmd=DISABLE,dccsrsprotmd=DISABLE,protocol=1+1_OPTM,rme=
      DISABLE,swreq=NR,ssi=WKG,actunit=WKG,actunitfe=NONE,
      actunitne=WKG,aps=NR,apsout=NR,pgrate=STM64,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

Multiplex Section/Line protection group (1:N)

The following example retrieves 1:N multiplex section/line protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-n001:123456::1x1;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-n001:protype=1xNOPT,wkg=1-1-#-#-13-1,protn=1-1-#-#-14-1,wtr=5,apspn=DEFAULT,
       dcclmsprotmd=DISABLE,dccsrsprotmd=DISABLE,protocol=1xN_BIDIR,rme=
      ENABLE,swreq=NR,actunit=WKG,actunitfe=WKG,
      actunitne=WKG,aps=NR,apsout=NR,pgrate=STM64,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

2 fibre MS-SPRing/BLSR protection group

The following example retrieves 2F MS-SPRing/BLSR protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-t001:123456::2F;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
```

```
   ”1-1-t001:protype=2F,east=1-1-#-#-13-1,west=1-1-#-#-14-1,rid=\”denver\”,wtr=5,
      blsrpn=DEFAULT,eswreq=NR,wswreq=NR,pgrate=OC192,eapschst=IDLE,eapsdnid=0,
      eapsprotn=0,eapssnid=0,eapssw=NR,elprotnsw=SUCCESS,ringndst=IDLE,wapschst=IDLE, wapsdnid=
;
```

4 fibre MS-SPRing/BLSR protection group

The following example retrieves 4F MS-SPRing/BLSR protection group information:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-f001:123456::4F;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ”1-1-f001:protype=4F,eastwkg=1-1-#-#-13-1,eastprotn=1-1-#-#-15-1,westwkg=1-1-#-#-
      14-1, westwkg=1-1-#-#-16-1,rid=\”denver\”,protnmode=LOOPBACK,swtr=5,wtr=5,
      blsrpn=DEFAULT,eswreq=NR,eactunit=WKG,wswreq=NR,
      wactunit=WKG,pgrate=OC192,eapschst=IDLE,eapsdnid=0,
      eapsprotn=0,eapssnid=0,eapssw=NR,elprotnsw=SUCCESS,ringndst=IDLE,wapschst=IDLE, wapsdnid=

;
```

Path protection group

Constituent Member List – Path protection group Examples: To help understand a PATHDRI examples that will be given below, it will be helpful to summarize some facts pertaining to DRIs. The first output line for a PATHDRI gives information about the DRI as a entity.

The following output line gives information about the constituent members. Even for a fixed rate path DRI, there will be two output lines. The constituent member that is described by the second line will have a rate equal to the DRIs rate.

A path DRI has two input legs, one referred to as the working leg and the other as the protection leg. It is possible for RTRV-PROTN-GRP to indicate some constituent members have been selected from the working leg while others have been selected from the protection leg. To determine whether or not all constituent members have been selected from the same leg, it is necessary to look at the *actunitaid* for each constituent member and determine if they are from the same leg.

Constituent Member List – Path protection group example for fixed mode :

```
RTRV-PROTN-GRP:LT-UNITE:1-1-#-#-10-1-25:123456::PATHDRI;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ”1-1-#-#-10-1-25:protype=PATHDRI,cst=STS12C,wkg=1-1-#-#-08-1-25,protn=1-1-#-#-06-
      1-1,ppgname=\”group018\”,htmr=100,wtr=5,dripathpn=path2,rme=DISABLE,snctype=NA”
   ”1-1-#-#-10-1-25,CML:cst=STS12C,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-25,wkgsiglvl=NONE,protnsiglvl=NONE”
;
```

Constituent Member List – Path protection group example for pipe mode: The examples shows an STS12 pipe with 10 constituent members.

```
RTRV-PROTN-GRP:LT-UNITE:1-1-#-#-10-1-25:123456::PATHDRI;
   LT-UNITE 05-05-31 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-1-25: protype=PATHDRI,htmr=100,rme=ENABLE,
      ppgname=\″group019\″,wtr=5,dripathpn=path2,cst=STS12C,wkg=1-1-#-#
-08-1-25,protn=1-1-#-#-06-1-1,snctype=NA″
   ″1-1-#-#-10-1-25,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-25,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-26,CML:cst=STS1,swreq=SF,actunit=PROTN,actu
nitaid=1-1-#-#-06-1-2,wkgsiglvl=SF,protnsiglvl=NONE″
   ″1-1-#-#-10-1-27,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-27,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-28,CML:cst=STS3C,swreq=SF,actunit=PROTN,actunitaid
=1-1-#-#-06-1-4,wkgsiglvl=SF,protnsiglvl=NONE″
   ″1-1-#-#-10-1-31,CML:cst=STS1,swreq=SD,actunit=PROTN,actunitaid=
1-1-#-#-06-1-7,wkgsiglvl=SD,protnsiglvl=NONE″
   ″1-1-#-#-10-1-32,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-32,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-33,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-33,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-34,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-34,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-35,CML:cst=STS1,swreq=NR,actunit=PROTN,actu
nitaid=1-1-#-#-06-1-11,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-10-1-36,CML:cst=STS1,swreq=NR,actunit=WKG,actuni
taid=1-1-#-#-08-1-36,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

Low Order Path Protection Group Example:

```
RTRV-PROTN-GRP:LT-UNITE:1-1-#-#-39-16-3-28:123456::PATHDRI;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-39-16-3-28:protype=PATHDRI,cst=LOVT1,wkg=1-1-#-#-39-14-3-28,\
    protn=1-1-#-#-39-15-3-28, htmr=75,wtr=5,dripathpn=path2,\
    rme=DISABLE,snctype=NA″
   ″1-1-#-#-39-16-3-28,CML:cst=LOVT1,swreq=NR,actunit=WKG,\
    actunitaid=1-1-#-#-39-14-3-28,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

Example for Path Protection Group Retrieval using *aid* and *aid1* for addressing via input tributaries:

In this example there exist two path protection groups on the same pair of input tributaries 1-1-#-#-8-1-25 and 1-1-#-#-6-1-25 forming an emulated selector bridge. The output tributary for one of these groups is 1-1-#-#-10-1-25 and for the other one it is 1-1-#-#-4-1-25.

Both groups will be reported after this request. Note, that the parameter values for both groups are identical.

```
RTRV-PROTN-GRP:LT-UNITE:1-1-#-#-8-1-25:123456::
               PATHDRI,,,1-1-#-#-6-1-25;
   LT-UNITE 98-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-10-1-25:protype=PATHDRI,cst=STS12C,wkg=1-1-#-#-08-1-25,
   protn=1-1-#-#-06-1-25,ppgname=\″group018\″,htmr=100,wtr=5,
   dripathpn=path2,rme=DISABLE,snctype=NA″
   ″1-1-#-#-10-1-25,CML:cst=STS12C,swreq=NR,actunit=WKG,
   actunitaid=1-1-#-#-08-1-25,wkgsiglvl=NONE,protnsiglvl=NONE″
   ″1-1-#-#-4-1-25:protype=PATHDRI,cst=STS12C,wkg=1-1-#-#-08-1-25,
   protn=1-1-#-#-06-1-25,ppgname=\″group018\″,htmr=100,wtr=5,
   dripathpn=path2,rme=DISABLE,snctype=NA″
   ″1-1-#-#-4-1-25,CML:cst=STS12C,swreq=NR,actunit=WKG,
   actunitaid=1-1-#-#-08-1-25,wkgsiglvl=NONE,protnsiglvl=NONE″
;
```

Error responses (general)

Validation rule: consistent *aid* and *protype*

If a range is specified for *aid* and *protype* is not consistent with that AID (for example, an AID of 1-1-fall and a protype value of 2F), then the following error response is returned.

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and protype`

Error responses (2-/4-fibre MS-SPRing/BLSR protection group)

Validation rule: consistent *aid* and *rid*

If both an *aid* and an *rid* are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Validation rule: *aid* range and *rid*

Validation rule: If a range is specified for *aid* and an *rid* is also specified as input parameters, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, no aid range in combination with rid`

Error responses (path protection group)

Validation rule: *ppgname* and low order path protections

If a *ppgname* is given as input parameter while the *aid* references a low order tributary, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, ppgname not supported for this transmission rate`

Validation rule: *ppgname* and *aid1*

If a *aid1* is given as input parameter and additionally a value is given for parameter *ppgname*, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, ppgname not allowed if aid1 is given`

Validation rule: No wildcarding on access via *aid* and *aid1*

If *aid* and *aid1* is given as input parameters for accessing via input tributaries and wildcarding is used for any of both, then the following error response is returned:

Error code:

`IDNV – Input, Data Not Valid, No wildcarding for access via input tribs`

Wrong system behavior

If 4F is used in the **RTRV-PROTN-GRP** command line although the used port AID belongs to a 2F protection group, or if 2F is used in the **RTRV-PROTN-GRP** command line although the used port AID belongs to a 4F protection group, the system shows a wrong behavior. Expected is an empty COMPLD report, however the command COMPLDs *and* the 2F/4F group is displayed.

Related TL1 messages

- “ENT-PROTN-GRP: Enter protection group” (p. 3-1745)
- “ED-PROTN-GRP: Edit protection group” (p. 3-803)
- “DLT-PROTN-GRP: Delete protection group” (p. 3-233)
- “OPR-PROTNSW: Operate protection switch” (p. 3-2057)
- “RLS-PROTNSW: Release protection switch” (p. 3-2203)

RTRV-PTF: Retrieve tributary with path termination function

The **RTRV-PTF** command retrieves the attributes of a tributary with path termination function (PTF).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-PTF**-*modifier:tid:aid:ctag[:::spec_block];*

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the tributary. The purpose is to distinguish the standard and type of the PTF.
   Type: Alphanumeric value set.
   Range:
   - ALL
   - STS for DS3 card
   - HOSTS1 for LOXC card
   - HOVC4 for LOXC card
   - TMASTS1 for TransMUX asynchronous port
   - TMSSTS1 for TransMUX synchronous port

   The value **ALL** is used to retrieve all path terminations providing lower order tributaries independent from the interface standard (for example, HOSTS1 and HOVC4).
   The following path terminations are not included:
   - path terminations related to DS3
   - path terminations related to TransMUX ports.
2. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
3. **aid** – Access Identifier
   AID of a PTF tributary. The keyword **all** is allowed on slot, port and tributary level.

The so called ″dynamic parameters″ is not reported, if a **`all`** is used at the tributary level. The ″dynamic parameters″ are:

- *ptrc, ptrcst, rsiglb, trailstate*

Type: AID

Range:

- PTNumAID for modifier: STS, HOSTS1, HOVC4, ALL
- TMuxAPTNumAID for modifier: TMASTS1
- TMuxSPTNumAID for modifier: HOSTS1, TMSSTS1

as defined in AID structure with exception of value **`all`**.

4. **`ctag`** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Spec block parameters

1. **`dyn`** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values all, usall or vall at the appropriate positions of the input AID. For input AIDs not using all, usall or vall this parameter is not allowed.

   The following output parameters of a PTF tributary are affected:

   - *ptrc, ptrcst, rsiglb, trailstate*

   Type: Alphanumeric value set.

   Range:

   - YES: Report dynamic parameters in the reply as well
   - NO (default): Do not report dynamic parameters in the reply.

Output format

If the NE fully complies with the **`RTRV-PTF`** request, the following normal completion response is returned:

```
sid date time
M  ctag COMPLD
   ″aid,aidtype::spec_block″
       ...
       ...
   ″aid,aidtype::spec_block″
;
```

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   The access identifier specifies the PTF tributary whose attributes are reported.
   Type: AID
   Range:
   - TNumAID for modifier: STS, HOSTS1, HOVC4, ALL
   - TMuxAPTNumAID for modifier: TMASTS1
   - TMuxSPTNumAID for modifier: HOSTS1, TMSSTS1

   as defined in AID structure with exception of value **`all`**.
2. **`aidtype`** – Access Identifier Type
   The *aidtype* specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range:
   - STS
   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   The parameter *brstintvl* specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST, then *brstintvl* has no affect to NE behavior.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.
   The parameter *brstintvl* is not supported for the following modifiers:
   - HOSTS1
   - HOVC4

- TMASTS1
- TMSSTS1

Option: If the parameter *brstintvl* is not supported, then the parameter is omitted.
Type: Integer.
Range: 2, 3, … 10 [seconds]
Default: 7 [seconds].

2. **`brstthr`** – Bursty Error Threshold

   The parameter *brstthr* specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS (path terminations related to physical T3 ports). But the parameters for near end performance monitoring on T3 related path terminations are kept for compatibility reasons.

   The parameter *brstthr* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMASTS1
   - TMSSTS1

   Option: If the parameter *brstthr* is not supported, then the parameter is omitted.
   Type: Integer.
   Range: 5, 10, 15, … 50 [%].
   Default: 30.

3. **`crslpbkstat`** – Cross-connect Loopback Status

   The parameter *crslpbkstat* reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035).

   The parameter *crslpbkstat* is not supported for the following modifiers:

   - HOSTS1
   - HOVC4
   - TMSSTS1

   Option: If the parameter *crslpbkstat* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: YES, for enabled loopback; NO, for disabled loopback

4. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to RTRV-PTF Output parameters.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.
   The parameter *degthr* is not supported for modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Integer.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
5. **`dexcthr`** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to RTRV-PTF Output parameters.
   Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.
   The parameter *dexcthr* is not supported for modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Integer.
   Range (as exponents of 10): –3; –4; –5.
   Default: –3.
6. **`exphptrc`** – Expected HP trace identifier
   This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if *ptrcrfmt* is set to 16.
   Limitation: The parameter is not supported for modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Hexadecimal number without a preceding “H”.
   Range: length: 32 hexadecimal digits; value: specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000

7. **`feppm`** – Far End Path PM Enable
   The parameter *feppm* specifies whether the performance monitoring for the far end is enabled or disabled for this tributary.
   This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
   Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the parameters for far-end performance monitoring on T3 related path terminations are kept for compatibility reasons.
   The parameter is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
8. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion
   The parameter *hptimais* shall specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   The parameter is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE.

   Default: The default value depends on interface standard specified by *ptfstd* for the PTF tributary. DISABLE, if *ptfstd* = SONET
9. **`hptimmon`** – HP Trace Identifier Mismatch Monitor
   This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   The parameter is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE

   Default: DISABLE.

10. **`inputsig`** – Tributary Input Signal Rate List
    This parameter specifies the input LO signal structure.
    The parameter *inputsig* is only supported for the following modifiers:
    - HOVC4
    - HOSTS1
    - TMSSTS1

    Option: If the parameter *inputsig* is not supported, then the parameter is omitted.
    Type: numeric value set
    Range: NX-NX-… (N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
    The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
    The possible values (SONET) are: 2811
    Restricted Range with AU3/AU4 Conversion cross-connections:
    If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *inputsig* is restricted to: 33.
    Restricted Range with TransMUX:
    If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.
    Default:
    - 6312 for SDH
    - 2811 for SONET
11. **`neppm`** – Near End Path PM Enable
    This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary.
    This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
    Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the possibility to enable near-end performance monitoring on T3 related path terminations is kept for compatibility reasons.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE.

    Default: DISABLE.
12. **`pmhpfeses`** – PM HP Far End SES Threshold
    This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *ptfstd* for the PTF tributary. If

ptfstd is not set to SDH then *pmhpfeses* has no affect to NE behavior.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Far-end performance monitoring is not available for modifier STS. But the possibility to enable far-end performance monitoring on T3 related path terminations is kept for compatibility reasons.

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

13. **pmhpneses** – PM HP Near End SES Threshold
This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end. The behavior depends on interface standard specified by *ptfstd* for the tributary. If *ptfstd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
This parameter is omitted for PTF entities which have been created due to a substructured SONET/SDH signal.
Limitation for PTF related modifiers: Near-end performance monitoring is not available for modifier STS. But the parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.

    The parameter is not supported for the following modifiers:

    - TMSSTS1
    - TMASTS1

    Option: If the parameter is not supported, then it should be omitted.
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000. Includes SDH/SONET range.
Default: 2400

14. **ptca** – Path TCA Profile Pointer
This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.

    TCA profile: The TCA profile type depends on the type and standard of the PTF, which is specified with the parameter *modifier* or *aidtype*:

    - PATH if *modifier* = TMASTS1

The parameter is not supported for the following modifiers:

- TMSSTS1
- TMASTS1
- HOVC4

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.
Range: 1 … 24 characters.
Default: DEFAULT0

15. **`ptfstd`** – Path Termination Standard
Standard of the tributary with Path Termination Function.
This is currently a read-only parameter.
Type: Alphanumeric value set.
Range:
   - SDH (only if *modifier* = HOVC4)
   - SONET (only if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1)

   Default:
   - SDH (if if *modifier* = HOVC4)
   - SONET (if *modifier* = STS, HOSTS1, TMASTS1, or TMSSTS1).
16. **`ptpn`** – Path Terminating ASA Profile Name
This parameter points to a tributary ASAP profile name *pfname* for the designated AID.
The parameter is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set. A hyphen ("-") is allowed in the name additionally.
Range: 1 … 24 characters
Default: DEFAULT
17. **`ptrc`** – Incoming Path Trace
This parameter reports the accepted higher order path trace identifier (J1). The length depends on the value of the parameter *ptrcrfmt*. The status of this value is contained in parameter *ptrcst*. If *ptrcst* indicates NORMAL, then *ptrc* contains a valid value. If *ptrcst* indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to not meeting the basic acceptance criteria or intra-NE communication problems, and the output parameter ptrc is omitted from the response.
The parameter is not supported for the following modifiers:
   - TMSSTS1

   Option: If the parameter is not supported, then it should be omitted.
Type: Hexadecimal

Range: the range depends on the parameter ptrcrfmt specified in RTRV-PTF Output parameters

- 2: hexadecimal digits if *ptrcrfmt*=2
- 32: hexadecimal digits if *ptrcrfmt*=16
- 128: hexadecimal digits if *ptrcrfmt*=64.

18. **`ptrcrfmt`** – Path Trace Read Format

    The parameter *ptrcrfmt* specifies the format of the path trace (J1 byte) to be received (sink).

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Numeric string.

    Range:

    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

19. **`ptrcst`** – Status of Incoming Path Trace

    This parameter shows the status of the value of the incoming Path Trace Identifier (J1) which is available with parameter *ptrc*.

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.

    Range:

    - NORMAL: *ptrc* contains a valid value
    - UNAVAILABLE: no valid value for *ptrc* could be obtained by the system
    - AIS: no valid value for *ptrc* could be obtained due to SF condition.

20. **`ptrcwfmt`** – Path Trace Write Format

    The parameter *ptrcwfmt* specifies the format of the path trace (J1 byte) to be sent (source).

    The parameter is not supported for the following modifiers:

    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Numeric string.

    Range:

    - 2: for non-specific repeating byte
    - 16: for 16 byte specific string
    - 64: for 64 byte string.

    Default: 2.

21. **`rsiglb`** – Received Signal Label
    This parameter reports the accepted trail signal label (C2). The parameter is not reported if the *trailst* parameter indicates a fail condition, i.e. has the value AIS or UNAVAILABLE.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range: 00, 01, …, FE, FF i.e. two hexadecimal digits.
22. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    This parameter specifies the determination mode of degraded defect for transmission.
    Limitation for PTF related to modifiers: Near-end performance monitoring is not available for modifier STS. The parameters for near-end performance monitoring on T3 related path terminations are kept for compatibility reasons.
    The parameter is not supported for the following modifiers:
    - TMSSTS1
    - TMASTS1
    - HOVC4
    - HOSTS1

    Option: If the parameter *sdsfmode* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: BURST, POISSON
    Default: The default value depends on interface standard specified by *ptfstd* for the tributary. POISSON, if *ptfstd* = SONET.
23. **`substr`** – Substructured
    The parameter *substr* indicates if the PTF is substructured.
    Additional Description for LOXC
    This parameter indicates if the PTF is configured for AU3/AU4 conversion cross-connections.
    The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created. A PTF, which is configured for AU3/AU4 conversion cannot be used to set up HO cross-connections.
    The parameter *substr* is only supported, if *modifier* = STS.
    If the parameter *substr* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - CNV: In use for HO side of a conversion cross-connection
    - NO (default): Not used for conversion cross-connections
24. **`tmonmd`** – Tributary Monitoring Mode
    This parameter specifies the monitoring mode for the tributary.

The parameter is not supported for the following modifiers:

- TMSSTS1

Option: If the parameter is not supported, then it should be omitted.
Type: Alphanumeric value set.
Range: MON, NMON (default).

25. **`trailst`** – Trail Status
    This parameter reports the accepted trail status value.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range:
    - AIS for Alarm Indication Signal in case of Trail Signal Fail Condition
    - UNAVAILABLE for unavailable
    - NORMAL
26. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Hexadecimal number without a preceding "H".
    Range: the range depends on the parameter ptrcwfmt specified in RTRV-PTF Output parameters
    - length: 32 hexadecimal digits, if *ptrcwfmt*=16; 128 hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt*=16; 64 byte string if *ptrcwfmt* = 64

    Default: 00. This implies also that *ptrcwfmt* is set to 2.
27. **`tslmmon`** – Trail Signal Label Mismatch Monitor
    The parameter *tslmmon* specifies the mode of the TSL mismatch detection.
    The parameter is not supported for the following modifiers:
    - TMSSTS1

    Option: If the parameter is not supported, then it should be omitted.
    Type: Alphanumeric value set.
    Range: DISABLE, ENABLE.
    Default: ENABLE.

28. **`unequiposig`** – Unequipped Output Signal List
    The parameter *unequiposig* specifies the output LO signal structure, independent whether the signal is connected or not. The parameter *unequiposig* is only supported for modifiers HOVC4. HOSTS1. and TMSSTS1.
    Option: If the parameter *unequiposig* is not supported, then the parameter is omitted.
    Type: numeric value set
    Range: NX-NX-…(N is the number of Xs. X can be 3, 12, or 11 representing VC3, VC12 or VT1.5).
    The possible values (SDH) are: 6312, 13-4212, 2112-13-2112, 4212-13, 23-2112, 13-2112-13, 2112-23, 33
    The possible values (SONET) are: 2811
    Restricted Range with AU3/AU4 Conversion cross-connections:
    If *modifier* = HOVC4 and the HO PTF is used for conversion cross-connections, then the range of *unequiposig* is restricted to: 33.
    Restricted Range with TransMUX:
    If *modifier* = TMSTS1, then the range of *inputsig* is restricted to: 2811.
    Default:
    - 6312 for SDH
    - 2811 for SONET

Detailed behavior description

The RTRV-PTF does not generate a REPT DBCHG message. The command retrieves all port parameter as described in section Output parameters.

Dependencies

brstintvl, *brstthr* and *sdsfmode* are not supported for modifiers HOVC4, HOSTS1,TMASTS1, and TMSSTS1 and will therefore not be reported in this case.

For all other PTFs:

- *brstintvl* and *brstthr* are only meaningful if *sdsfmode* is set to BURST.
- *degthr* and *dexcthr* are only meaningful if *sdsfmode* is set to POISSON.

Parameters which are not meaningful in the current provisioning status are reported with their default values, nevertheless.

Behavior in the case of Conversion Cross-Connections

The **`RTRV-PTF`** command does not return data for the HO sides of conversion cross-connections at synchronous optical and electrical ports.

PTFs exist at the LO side of conversion cross-connections, and if the HO side is at DS3 ports, also at the HO side of conversion cross-connections.

No PTFs exist at the HO side of conversion cross-connections, when the HO side is at a synchronous optical or electrical port.

Example input/output

```
RTRV-PTF-STS:LT-UNITE:1-1-#-#-21-4-1:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-21-4-1,STS::brstintvl=7,brstthr=30,crslpbkstat=NO,degthr=-6,dexcthr=-
      3,exphptrc=89000000000000000000000000000000,feppm=DISABLE,hptimais=DISABLE,hptimmon=DI
;

RTRV-PTF-HOVC4:LT-UNITE:1-1-#-#-22-4-1:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-22-4-
      1,HOVC4::exphptrc=89000000000000000000000000000000,feppm=DISABLE,inputsig=6312,hptimai
;

RTRV-PTF-TMASTS1:LT-UNITE:1-1-#-#-22-tma4-1:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-22-tma4-1,TMASTS1::degthr=-6, dexcthr=-
      3,exphptrc=89000000000000000000000000000000,feppm=DISABLE,hptimais=ENABLE,hptimmon=DIS
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-PTF** command, too.

Validation rule: Check CCB against *ptfstd*

Check wether the CCB fits to ptfstd of the tributary.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with modifier=VC is made on a port which is provisioned as ptfstd=SONET.

Related TL1 messages

- “ED-PTF: Edit tributary with Path Termination Function (PTF)” (p. 3-830)

RTRV-RC-GENERAL: Retrieve general routing controller settings

The **RTRV-RC-GENERAL** command retrieves the general settings of the network element routing controller.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-RC-GENERAL**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*.

Output format

If the **RTRV-RC-GENERAL** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   "spec_block"
;
```

Output parameters

Spec block parameters

1. **hellointerval** – Hello Interval
   This parameter is used to indicate the time (in seconds) between two succeeding "Hello" messages to be sent to a neighbor RC.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 10.
2. **deadintv** – Dead Interval
   This parameter is used to indicate the time (in seconds) that the system waits for the reception of "Hello" PDU from a neighbor RC before the neighbor RC is considered to be dead.
   Type: Numerical.
   Range: 1 … 65535.
   Default: 30.

Detailed behavior description

The command retrieves the routing controller general settings.

Example input/output

The following example shows the result of **RTRV-RC-GENERAL**:

```
RTRV-RC-GENERAL:NE-UNITE::123456;
   NE-UNITE 06-11-06 11:01:00
M  123456 COMPLD
   "hellointv=10, deadintv=30"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-RC-GENERAL** command.

Validation rule: RC must be created

If the RC is not created (through **ACT-L1ROUTING**), then the command is denied.

Error code: `IDNV – Input, Data Not Valid, RC not created`

Related TL1 messages

- "ED-RC-GENERAL: Edit general routing controller settings" (p. 3-853)

RTRV-RC-NEIGHBOR: Retrieve all neighboring routing controllers

The **RTRV-RC-NEIGHBOR** command is used to retrieve all neighboring routing controllers that have been provisioned using the **ENT-RC-NEIGHBOR** command.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-RC-NEIGHBOR**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Output format

If the **RTRV-RC-NEIGHBOR** command is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "[rate]:thru[,lpaid]:spec_block"
   ...
   "[rate]:thru[,lpaid]:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output Parameter section for the “RTRV-HDR: Retrieve header” (p. 3-2417) command.

Output parameters

Spec block parameters

1. **`remrcid`** – Remote Routing Controller Identifier
   This parameter is used to indicate the Id of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
2. **`remrcscn`** – Remote Routing Controller IP Address
   This parameter is used to indicate the IP address of the remote routing controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
3. **`remrcraid`** – Remote Routing Controller Routing Aread ID
   This parameter is used to indicate the routing area ID of the level 0 area of a neighboring route controller.
   Type: Four dot separated numeric strings.
   Range: Each numeric string: 0 .. 255.
   0.0.0.0 is not allowed.
4. **`remrcOspfFlooding`** – Remote Routing Controller OSPF Flooding
   This parameter indicates whether OSPF flooding is enabled to the remote RC or not.
   Type: Alphanumeric value set.
   Range: DISABLE or ENABLE.

Detailed behavior description

The command shall retrieve all routing controller neighbors.

Example input/output

The following example shows the result of a **RTRV-RC-NEIGHBOR** command:

```
RTRV-RC-NEIGHBOR:NE-UNITE::123456;
   NE-UNITE 06-11-06 11:01:00
M  123456 COMPLD
   ″remrcid=1.2.3.4, remrcscn=124.24.12.56, remrcraid=1.2.3.4,
      remrcOspfFlooding=ENABLE″
   ″remrcid=5.2.3.1, remrcscn=132.12.96.2, remrcraid=17.2.5.9,
      remrcOspfFlooding=DISABLE″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-RC-NEIGHBOR** command.

Validation rule: RC must be created

If the RC is not created (via ACT-L1ROUTING), then the command is denied.

Error code: `IDNV – Input, Data Not Valid, RC not created`

Related TL1 messages

- "ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)
- "DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers" (p. 3-247)

RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group

The **RTRV-RESALLOC-VCG** command retrieves the resource allocation of a ES64U in a slot. The **RTRV-RESALLOC-VCG** command is needed to provide the user an overview about the backplane fragmentation where blocking configurations are present.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-RESALLOC-VCG**:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   The AID of a circuit pack.
   Type: CpAID.
   The values **usall** for Simple_UnivSlotAID, **all** for Simple_UnivSlotAID, **all** for Simple_ShelfAID, and **all** for BayAID are excluded.
   Default: None.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the NE fully complies with the **RTRV-RESALLOC-VCG** request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "sts1equi:resalloc"
          ...
          ...
   "sts1equi:resalloc"
     <one output line for every STS-1 equivalent, together 384 lines.>
;
```

Output parameters

Command parameters

1. **`sts1equi`** – STS-1 Equivalent
   The STS-1 equivalent describes a single backplane bandwidth element from the physical backplane structure, which is represented in STS-1 (respectively HO-VC-3) granularity.
   The STS-1 equivalent reflects that the ES64U backplane bandwidth is always split in two parts of each 192 STS-1 or VC-3.
   Type: String.
   Range: Combined string ***x***'***S***'***yyy***, where
   x = 1 | 2, for part 1 or part 2 of the backplane bandwidth segment;
   yyy = {1…192}, for the number of the STS-1 equivalents.
   "1S" or "2S" specify the first or second bandwidth segment in total circuit pack bandwidth, that means, "1S" denotes the first 192 STS-1 or VC-3 and "2S" the second 192 STS-1 or VC-3 of anES64U.
   Default: None.
2. **`resalloc`** – Resource Allocation
   The resource allocation shall list the physical backplane structure in STS-1 (respectively HO-VC-3) granularity. For each backplane bandwidth element the allocated VCGTRIB instance shall be referenced. Unused backplane bandwidth elements shall be listed as well.
   It should be noted that a contiguous concatenated VCGTRIB instance has multiple STS-1 or VC-3 bandwidth (e.g. a single STS-3c is then listed three times for this STS-1 granularity). A VC-4 is handled like a STS-3c.
   Type: String.
   Range: Combined string ***<tribaid>*** | "unused".
   Default: "unused".

Detailed behavior description

The **RTRV-RESALLOC-VCG** does not generate a **REPT DBCHG** message. The command retrieves the resource alloaction as described in section" Output parameters".

Example input/output

Example output for SONET *istd*, i.e. listing in STS-1 granularity for two 10G bandwidth segments 1S and 2S with each 192 STS-1 tributaries:

```
RTRV-RESALLOC-VCG:UNITE-NE:1-1-#-#-22-cp:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″1S001:1-1-#-#-22-v1-1″
   ″1S002:1-1-#-#-22-v1-3″
   ″1S003:unused″
   ″1S004:unused″
   ″1S005:1-1-#-#-22-v1-2″
```

```
   "1S006:1-1-#-#-22-v8-1"
   "1S007:1-1-#-#-22-v8-3"
   "1S008:unused"
   "1S009:1-1-#-#-22-v8-2"
   "1S010:1-1-#-#-22-v4-1"
...
   "1S192:1-1-#-#-22-v1-4"
   "2S001:1-1-#-#-22-v205-8"
   "2S002:1-1-#-#-22-v205-12"
   "2S003:1-1-#-#-22-v205-6"
   "2S004:1-1-#-#-22-v205-3"
   "2S005:1-1-#-#-22-v205-2"
   "2S006:1-1-#-#-22-v205-5"
   "2S007:1-1-#-#-22-v205-11"
   "2S008:1-1-#-#-22-v205-9"
   "2S009:1-1-#-#-22-v205-7"
   "2S010:1-1-#-#-22-v205-10"
   "2S011:1-1-#-#-22-v205-4"
   "2S012:1-1-#-#-22-v205-1"
   "2S013:unused"
...
   "2S181:1-1-#-#-22-v207-7"
   "2S182:1-1-#-#-22-v207-4"
   "2S183:1-1-#-#-22-v207-2"
   "2S184:1-1-#-#-22-v207-1"
   "2S185:1-1-#-#-22-v207-9"
   "2S186:1-1-#-#-22-v207-6"
   "2S187:1-1-#-#-22-v207-5"
   "2S188:1-1-#-#-22-v207-3"
   "2S189:1-1-#-#-22-v207-8"
   "2S190:unused"
   "2S191:unused"
   "2S192:unused"
;
```

The example output above shows one time a STS-1-4v (v1), one time a STS-1-1v (v4), STS-1-12v (v205), one time a STS-1-9v (v207) and one time a STS-1-3v (v8).

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`RTRV-VCG-TRANS`** command, too.

Validation rule: Command only allowed for ES64U

If the command is not applied to a ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Command not allowed for ES64U with switch capacity 0

If the switch capacity *es64ucap* for a ES64U pack is 0, then the command is denied.

Error code:

`IENE – Input, Entity Not Exists, Command not allowed for an inactive ES64U pack`

Validation rule: Command not allowed for ES64U protecting pack

If the command is applied to an ES64U protecting pack in an equipment protection, then the command is denied.

Error code:

`IENE – Input, Entity Not Exists, Command not allowed for a protecting ES64U pack`

Related TL1 messages

- “DLT-VCG – Enter virtual concatenation group” (p. 3-277)
- “ENT-VCG – Enter virtual concatenation group” (p. 3-1983)
- “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078)

RTRV-RSVP: Retrieve RSVP parameters

The **RTRV-RSVP** command retrieves RSVP related parameters applicable for UNI or ENNI signaling on system level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-RSVP**-*modifier*:*tid*::*ctag*;

Input parameters

To retrieve RSVP parameters related to UNI signalling the TL1 command **RTRV-RSVP** is only valid with modifier UNI, that means, **RTRV-RSVP-UNI**.

Command parameters

1. **modifier**
   The parameter modifier indicates the communication function entity for which the RSVP parameters apply.
   Type: Alphanumeric value set.
   Range: UNI or ENNI.
2. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*.

Output format

If the **RTRV-RSVP** request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   ":spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Spec block parameters

1. **`hellointerval`** – Hello Interval
   This parameter is used to indicate the time between two succeeding HELLO messages to be sent to any neighbor in hello interworkings.
   Type: Numeric string.
   Range: 1 … 60
2. **`maxhelloretransmit`** – Maximum Hello Retransmissions
   This parameter is used to indicate the number of Hello Intervals without any HELLO_REQUEST or HELLO_ACK object received from a neighbor after which communication to this neighbor is considered as lost.
   Type: Numeric string.
   Range: 1 to 10.
   Default: 3
3. **`srefreshtimer`** – Srefresh Timer
   This parameter is used to indicate the maximum time between two consecutive SREFRESH requests for a UNI or network connection towards affected neighbors srefresh interworkings.
   Type: Numeric string.
   Range: 10 … 180 indicating the Srefresh time in seconds.
   Default: 15.

Detailed behavior description

The command shall retrieve the RSVP related parameters from the system for the node specified by the command modifier.

Example input/output

The following example shows the result of **`RTRV-RSVP`** for UNI:

```
RTRV-RSVP-UNI:NE-UNITE::123456;
   NE-UNITE 06-06-30 11:01:00
M  123456 COMPLD
   ":hellointerval=5,maxhelloretransmit=3,srefreshtimer=15"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-RSVP** command.

Validation rule: ENNI node must be created

If the command is executed with *modifier* = ENNI while the ENNI Node is not created (by provisioning of the *sigPCIP_Address* in **ENT-ULS**), then the command is denied.

Error code: `SNVS – Status, Not in Valid State, E-NNI Node not created`

Validation rule: UNI node must be created

If the command is executed with *modifier* = UNI while the UNI Node is not created, then the command is denied.

Error code:
`SNVS – Status, Not in Valid State, UNI Node IP Address not provisioned`

Related TL1 messages

- "ED-RSVP: Modify RSVP parameters" (p. 3-857)

RTRV-SCN-NEIGHBOR: Retrieve SCN neighbor

The **RTRV-SCN-NEIGHBOR** command retrieves entries from the routing table of the SCN stack.

Note that the information on neighbors given in the **RTRV-SCN-NEIGHBOR** command is related to the SCN, i.e. to the control plane of ONNS, whereas the neighbor information given in the RTRV-rr commands relates to the transmission plane of ONNS. In general the information given in RTRV-SCN-NEIGHBOR includes DCC and (ISIS) LAN neighbors which are assigned to the SCN.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-SCN-NEIGHBOR**:*tid*::*ctag*:::;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: TID.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
   Type: CTAG.

Output format

If the RTRV-SCN-NEIGHBOR request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   "[rate]:thru[,lpaid]:spec_block"
   ...
   "[rate]:thru[,lpaid]:spec_block"
;
```

Output parameters

Command parameters

1. **`rate`** – DCC Port Rate
   This parameter is only available for an SCN neighbor on DCC and provides the port rate as used in RTRV-rr.
   Type: value set.
   Range:
   - STM1
   - STM1E
   - STM4
   - STM16
   - STM64
   - STM256
   - OC3
   - OC12
   - OC48
   - OC192
   - OC768

   Default: n/a – as determined by the port type.
2. **`thru`** – Connected Through
   The type of connectivity between the NE and its SCN neighbor.
   Type: Value set.
   Range: DCC, LAN.
   Default: n/a – as determined by the type of connection.
3. **`lpaid`** – Local Port AID
   This parameter provides the SPortAID of the optical port in case thru=DCC or the LanPortAID in case thru=LAN.
   Type: AID
   Range: SPortAID in case thru=DCC or LanPortAID in case thru=LAN as defined in section AID Structure.
   Note: Since LAN1 can not be assigned to the SCN it will never appear in the output.

Spec block parameters

1. **`dcctype`**
   This parameter specifies, whether MS(LINE)-DCC or RS(SECTION)-DCC is used for IP communication with the neighbor. This parameter is only available when thru=DCC.
   Type: Alphanumeric value set.
   Range: SECTION, LINE
2. **`neighbor_ip`** – Neighbor IP Address
   This parameter is used to retrieve the IPv4 address of the LAN or DCC neighbor.
   Type: Four dot separated numeric strings or empty string.
   Range: Each numeric string: 0 .. 255 or empty string.
   Note: The empty string is shown when the IP address can not be retrieved. In output direction the empty string is represented as \″\″.
3. **`neighbor_sid`** – Neighbor SID
   This parameter is used to display the SID of the neighbor.
   Note: Since the SID (MAC address) is unique it might be used to detect misconfigurations of duplicate IP addresses.
   Type: Hexadecimal number.
   Range: 000000000000 – FFFFFFFFFFFF.

Detailed behavior description

The command shall retrieve the *spec_block* parameters as described in the Output parameter section.

A neighbor entry shall not appear in the report for LANs and DCC channels allocated to an OSI_node_AID or the UNI_node_AID or the ENNI_node_AID.

Behavior if no neighbors present

An empty spec block is returned, if no neighbors are present.

Example input/output

The following example shows the result of a RTRV-SCN-NEIGHBOR command:

```
RTRV-SCN-NEIGHBOR:LT-UNITE::123456:::;
   LT-UNITE 00-11-01 08:00:00
M  123456 COMPLD
   ″OC48:DCC,1-1-#-#-1-1:dcctype=LINE,neighbor_ip=192.168.200.1″
   ...
   ″:LAN,lan2: neighbor_ip=192.168.200.4″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-SCN-NEIGHBOR** command.

RTRV-SPANSWS: Retrieve SPAN switch state

General

The **RTRV-SPANSWS** command requests the NE to report the current switch states of an MS-SPRing TOP+EX ring protection group related to the cross-connections either being added or dropped, ingressing or egressing ectra traffic cross-connections and addressed to a specified side of the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-SPANSWS***:tid:[aid]:ctag::side,rcrole,[rid];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an service port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: FourFibre_ProtGrpAID or SPortAID.
   SPortAID identifies the protection group only, it will not be evaluated if it is located on east or west side. For this purpose specify the *side* parameter.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
4. **side** – side identifier of protection group
   This parameter specifies the ring side of the ring protection group.
   Type: Alphanumeric value set.
   Range: EAST, WEST

5. **`rcrole`** – ring cross-connection role
   This parameter specifies the cross-connection role of the ring node.
   Type: Alphanumeric value set.
   Range:
   - ADD, service cross-connections added to the ring
   - DROP, service cross-connections dropped from the ring
   - INGRESSING, ingressing extra traffic cross-connections
   - EGRESSING, egressing extra traffic cross-connections
6. **`rid`** – Ring Identification
   The parameter specifies the ring identification name. Storage and retrieval of the rid is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: Text string of chars allowed for TID strings
   Range: 1 to 15 characters
   Option: If *aid* is specified, then *rid* can be omitted.

Output format

The general format of a RTRV-SPANSWS message is as follows:

```
   tid date time
M  123456 COMPLD
   "aid:rid:side,rcrole"
   "fsts1pos:rate:protsiglvl"
   "fsts1pos:rate:protsiglvl"
       ...
       ...
   "fsts1pos:rate:protsiglvl"
;
```

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter titles the listing with the AID of the reported protection group.
   Type: AID
   Range: FourFibre_ProtGrpAID
2. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: Text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
3. **`side`** – side identifier of protection group
   This parameter titles the listing with the reported ring side of the ring protection group.
   Type: Alphanumeric value set.
   Range: EAST, WEST.
4. **`rcrole`** – ring cross-connection role
   This parameter titles the listing with the reported cross-connection role of the ring node.
   Type: Alphanumeric value set.
   Range:
   - ADD, service cross-connections added to the ring
   - DROP, service cross-connections dropped from the ring
   - INGRESSING, ingressing extra traffic cross-connections
   - EGRESSING, egressing extra traffic cross-connections
5. **`fsts1pos`** – first STS-1 tributary position
   The parameter specifies the first ’STS1NumAID’ of the listed constituent signal of the protection group, i.e. the first STS-1 position of the tributary. Each row of the list of reported constituent signals shall start with this parameter. The parameter *fsts1pos* is comparable to the parameters *in_aid* and *out_aid* of the RTRV-CRS output.
   Type: AID
   Range: STS1NumAID, the part Simple_STS1NumAID of this STS1NumAID is currently limited to 1 … 192.

6. **`rate`** – cross-connection rate
   The parameter specifies the cross-connection rate.
   Type: Alphanumeric value set.
   Range: VC3, VC4, VC44C, VC416C, VC464C.
   All rates are shown as SDH rates here. Internal conversion may be needed, see chapter 'Detailed behavior'. Refer also to *rate* in output parameters section of "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759).
7. **`protsiglvl`** – protection signal level
   The parameter specifies the current protection state of locally added or dropped service cross-connection for the TOP+EX application. The information includes the current state of either the switches or the bridges of one side of a TOP+EX protection group. It does not represent the switch request state.
   Type: Alphanumeric value set.

   Range for cross-connections fulfilling the add role:
   - AFFBR, affected and bridged
   - AFFNOBR, affected but not bridged
   - NOTAFF, not affected OR bridging not allowed in general
   - UNKNOWN, unknown, ring communication in progress

   Range for cross-connections fulfilling the drop role:
   - AFFSW, affected and switched
   - AFFNOSW, affected but not switched
   - NOTAFF, not affected OR switching not allowed in general
   - UNKNOWN, unknown, ring communication in progress.

   Range for extra traffic cross-connections (ingressing and egressing):
   - AFFRSTBL, affected and reestablished
   - AFFPREMT, affected and preempted
   - AFFNORST, affected, but not reestablished
   - NOTAFF, not affected.

Detailed behavior description

Source and presentation of rate

The source of the parameter *rate* is the cross-connection rate, i.e. that value which had been provisioned before by means of the ENT-CRS-*modifier* command. The presentation of the parameter rate is always the SDH nomenclature. For the case that the cross-connection had been entered in the SONET nomenclature a conversion to SDH nomenclature is performed. Keep in mind that only service traffic cross-connections are presented.

Example input/output

The following example shows a RTRV-SPANSWS report:

```
   LT-UNITE 99-01-10 08:12:00
M  123456 COMPLD
   "1-1-f001:\"ring01\":EAST,DROP"
   "1-1-#-#-1-1-6:VC3:AFFSW"
   "1-1-#-#-1-1-7:VC4:AFFSW"
   "1-1-#-#-1-1-10:VC44C:AFFSW"
   "1-1-#-#-1-1-22:VC416C:AFFNOSW"
   "1-1-#-#-1-1-103:VC464C:NOTAFF"
;
```

Error responses

Validation rule: RTRV-SPANSWS only for TOPEX

The system shall reject all RTRV-SPANSWS commands if the addressed protection group is not provisioned for MS-SPRing TOP+EX (*protnmode* /= TOPEX)

Error code:

`SNVS – Status, Not in Valid State, command supported for MS-SPRing TOPEX only`

Validation rule: RTRV-SPANSW needs functioning XC pack

The command is denied if the XC pack is not present or in failed state.

Error code: `SPFA – Status, Protection unit FAiled` The **`RTRV-SPANSW`** command uses the protection state machine running on the XC pack.

Related TL1 messages

- "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)

RTRV-STATE-EQPT (CP): Retrieve equipment state (circuit pack)

The **RTRV-STATE-EQPT** command retrieves the state information for circuit packs in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STATE-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   The AID of a circuit pack.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for circuit pack in any slot
   - LoxcAID ::= UnivSlotAID-loxc for LOXC pack in universal slots.

   The keyword **xcall** is allowed for Simple_XCSlotAID
   The keyword **ctlall** is allowed for Simple_CTLSlotAID
   The keyword **all** is allowed for Simple_SlotAID
   Refer to the AID syntax description.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **RTRV-STATE-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:spec_block"
       ...
       ...
```

```
   "aid:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command Parameter

1. **aid** – Access Identifier
   The AID of a circuit pack.
   Type: AID
   Range:
   - CpAID ::= SlotAID-cp for circuit pack in any slot
   - LoxcAID ::= UnivSlotAID-loxc for LOXC pack in universal slots.

   The keyword **usall** is allowed for Simple_UnivSlotAID
   The keyword **xcall** is allowed for Simple_XCSlotAID
   The keyword **ctlall** is allowed for Simple_CTLSlotAID
   The keyword **all** is allowed for Simple_SlotAID
   Refer to the AID syntax description.

Spec block parameters

1. **opstate** – Operational State
   Each circuit pack has a retrievable attribute indicating whether circuit pack is operational for the system internal communication.
   Type: Alphanumeric value set.
   Range:
   - OS_ENABLED
   - OS_DISABLED (default).

Detailed behavior description

The **RTRV-STATE-EQPT** command on system level (AID = system) retrieves the current mode of the system.

Besides the normal operation mode, the system provides the maintenance to:

- handle error conditions the system cannot manage autonomously
- apply privileged commands

In the maintenance mode acceptance of commands is restricted in the following way:

- commands allowed in maintenance and in normal mode:
  - get any provisionable parameter: any RTRV command
  - get any non-provisionable parameter: any RTRV command

  - set interface standard default mode: **ED-EQPT (system)**
  - set tributary default mode: **ED-EQPT (system)**
  - set maximum switch capacity: **ED-EQPT (system)**
  - set port auto-to-mon delay: **ED-EQPT (system)**
  - set cross connect setup: **ED-EQPT (system)**
  - set port class default: **ED-EQPT (system)**
  - database download, SW download: **CPY-MEM**, **COPY-RFILE**
  - activate downloaded database: **APPLY** (database)
  - set time: **ED-DAT**
  - set date: **ED-DAT**
  - set LAN parameter: **ENT-FECOM-LAN** (note: restrictions depend on parameter base, details see command description)
  - set IP parameter: **ENT-ULS** (note: restrictions depend on parameter base, details see command description)
- commands allowed in maintenance mode only
  - set TID: **SET-SID**
  - set SONET/SDH synchronization default mode: **ED-EQPT (system)**
  - set UPSR port alarming mode: **ED-EQPT (system)**
  - database restoration (download + installation): **CPY-MEM**, **COPY-RFILE**
  - leave maintenance mode (accept data base): **ED-STATE-EQPT (system)**
  - set IP addresses and subnet mask: **ENT-FECOM-LAN**, **ENT-ULS** (note: restrictions depend on parameter base, details see command description)

Accepting the database will be a direct result of leaving the maintenance mode. The system shall autonomously leave maintenance mode without accepting the database after a database download from the management interface. The **RTRV-STATE-EQPT** command does not generate a **REPT DBCHG** message.

Example input/output

```
RTRV-STATE-EQPT:UNITE-NE:1-1-#-#-2-cp:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ”1-1-#-#-2-cp:opstate=OS_ENABLE”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-STATE-EQPT** command.

Related TL1 messages

- "ED-STATE-EQPT (system): Edit equipment state" (p. 3-866)
- "ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)

References

GR-833-CORE

RTRV-STATE-EQPT (slot): Retrieve equipment state (slot)

The **RTRV-STATE-EQPT** command retrieves the state information for slots in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-STATE-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a slot.
   Type: AID
   Range:
   - SlotAID
   - UnivSlotAID
   - XCSlotAID
   - CTLSlotAID.

   The keyword **usall** is allowed for Simple_UnivSlotAID.
   The keyword **xcall** is allowed for Simple_XCSlotAID.
   The keyword **ctlall** is allowed for Simple_CTLSlotAID.
   The keyword **all** is allowed for Simple_SlotAID.
   The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **RTRV-STATE-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:spec_block"
```

```
       ...
       ...
   ″aid:spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

aid – Access Identifier

See definition of output parameter *aid* in “RTRV-EQPT (slot): Retrieve equipment (slot)” (p. 3-2386).

Spec block parameters

1. **`holdst`** – Holder Status
   Each provisionable slot has an attribute indicating its equipage state.
   Type: Alphanumeric value set.
   Range:
   - EMPTY. The system uses the value ″EMPTY″ if no circuit pack is equipped in that slot.
   - VALID. The system shall use the value ″Valid″ if a circuit pack is equipped in that slot as provisioned.
   - UNEXPECTED. The system shall use the value ″Unexpected″ if a different circuit pack is equipped in that slot as provisioned or if the circuit pack is not acceptable for this slot.
   - UNKNOWN. The system shall use the value ″Unknown″ if a circuit pack is equipped in that slot with non-readable functional type or illegal values for the functional type.

   Default: EMPTY.

Detailed behavior description

The system retrieves the output parameters as described above.

Example input/output

```
RTRV-STATE-EQPT:UNITE-NE:1-1-#-#-2:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″1-1-#-#-2:holdst=VALID″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **ED-STATE-EQPT** command.

Related TL1 messages

- "ED-STATE-EQPT (system): Edit equipment state" (p. 3-866)
- "ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)

References

GR-833-CORE

RTRV-STATE-EQPT (socket): Retrieve equipment state

The **RTRV-STATE-EQPT** command shall retrieve the state information for sockets in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STATE-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keyword *usall* or *all* is allowed for Simple_UnivSlotAID. The keyword *scall* is allowed for Simple_SocketAID. The keyword **all** is not allowed on bay/shelf level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **RTRV-STATE-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_ block]″
   ... ...
   ″aid[:spec_ block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   The AID of a socket.
   Type: AID.
   Range: SocketAID. The keywords *all* / *usall* / *scall* shall are not allowed on any level.

Spec block parameters

1. **`holdst`** – Holder Status
   Each provisionable socket shall have an attribute indicating its equipage state.
   Type: Alphanumeric value set.
   Range:
   - EMPTY, the system shall use the value ″EMPTY″ if no module is equipped in that socket.
   - VALID, the system shall use the value ″Valid″ if a module is equipped in that socket as provisioned.
   - UNEXPECTED, the system shall use the value ″Unexpected″ if a different module is equipped in that socket as provisioned or if the module is not acceptable for this socket.
   - UNKNOWN, the system shall use the value ″Unknown″ if a module is equipped in that socket with non-readable functional type or illegal values for the functional type.

   For further details w.r.t. state transitions for holder status – see SRD_Configuration-Management.
   Default: EMPTY.

Detailed behavior description

The system shall retrieve the output parameters as described above.

Example input/output

This example shows a RTRV-STATE-EQPT command for a socket:

```
RTRV-STATE-EQPT:UNITE- NE:1-1-#-#-5-sc2:123456;

   UNITE- NE 00-11-03 08:00:00
M  123456 COMPLD
   ″1- 1-#-#-5-sc2:holdst=VALID″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`ED-STATE-EQPT`** command.

Related TL1 messages

- “ED-EQPT (CP): Edit equipment (circuit pack)” (p. 3-349)
- “ENT-EQPT (slot): Enter equipment (slot)” (p. 3-1547)

References

GR-833-CORE

RTRV-STATE-EQPT (system): Retrieve equipment state (system)

The **RTRV-STATE-EQPT** command retrieves the state information on system (NE) level.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STATE-EQPT***:tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system; the usage of the keyword **all** on system level is not allowed.
   Default: system
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

If the **RTRV-STATE-EQPT** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "system:spec_block"
       ...
       ...
   "system:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

1. **`aid`** – Access Identifier
   The aid of the system itself.
   Type: AID
   Range: system; the usage of the keyword **`all`** on system level is not allowed.
   Default: system

Spec block parameters

1. **`commitstate`** – SW Commitment Status
   The system provides an indication of the SW generic commitment status.
   Type: Alphanumeric value set.

   Range:

   - PENDING = the SW generic is in trial phase, switch back to previous generic possible
   - COMMITTED = the SW generic is comitted, no switch back possible.

2. **`dbstate`** – Data Base Change State
   The system provides an indication of the validity of the active data base.
   Type: Alphanumeric value set.

   Range:

   - EMPTY = data base state empty.
   - NEW = data base state new
   - VALID = database state valid
   - UNKNOWN = database state is not known.

   Default: UNKNOWN.

3. **`mcond`** – Maintenance Condition
   This parameter denotes whether the system condition is in maintenance condition or not.
   Type: Alphanumeric value set.

   Range:

   - Y = maintenance mode
   - N = normal mode.

Detailed behavior description

The **`RTRV-STATE-EQPT`** command on system level (AID = system) retrieves the current mode of the system.

The system provides the maintenance mode besides the normal operation mode in order to – handle error conditions the system can not autonomously manage – apply privileged commands.

In this maintenance mode acceptance of commands is restricted in the following way:

- commands allowed in maintenance AND normal mode:
  - get any provisionable parameter: any RTRV command
  - get any non-provisionable parameter: any RTRV command
  - set interface standard default mode: ED-EQPT (system)
  - set tributary default mode: ED-EQPT (system)
  - set maximum switch capacity: ED-EQPT (system)
  - ED-EQPT (system)
  - database download to restore empty/corrupt DBCPY-MEM
  - set time: ED-DAT
  - set date: ED-DAT
  - set IP parameter: ENT-ULS
    (note: restrictions depend on parameter base, details see command)
  - set LAN parameter: ENT-FECOM-LAN
    (note: restrictions depend on parameter base, details see command).
- commands ONLY allowed in maintenance mode:
  - set TID SET-SID
  - set SONET/SDH synchronisation default
  - leave maintenance mode (accept data base): ED-STATE-EQPT (system)

  Accepting the database will be a direct result of leaving the maintenance mode.

The system autonomously leaves maintenance mode without accepting the data base after a data base download from the management interface.

The **RTRV-STATE-EQPT** command does not generate a **REPT DBCHG** message.

Example input/output

```
RTRV-STATE-EQPT:UNITE-NE:system:123456;

   UNITE-NE 00-11-03 08:00:00
M  123456 COMPLD
   ″system:dbstate=EMPTY″
   ″system:mcond=Y″
;
```

Error responses

Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command Error responses section.

Related TL1 messages

- "ED-STATE-EQPT (system): Edit equipment state" (p. 3-866)
- "ED-EQPT (system): Edit equipment (system)" (p. 3-391)
- "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)

RTRV-STM1E: Retrieve STM1E port

The command **RTRV-STM1E** is used to retrieve all parameters related to the electrical STM1E port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STM1E**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an STM1E port.
   Type: AID.
   Range: SPortAID of STM1E port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-STM1E command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Access Identifier of STM1 port.
   Type: AID.
   Range: SPortAID of STM1E port.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **`femspm`** – Far-end Multiplex Section PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.

Range:

- ENABLE , set a forced DUS/DNU
- DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
    Default: 14.

13. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.

Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

15. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
16. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
This identifies the maximum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
maxNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC4 (default), VC3.
17. **`minNNXCRate`** – Minimum NN Cross Connect Rate
This identifies the minimum rate at which cross-connections are allowed for ports of *portClass* INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
minNNXCRate is introduced specifically for use by ONNS application.
Type: Alphanumeric value set.
Range: VC3 (default), VC4.
18. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.

Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

19. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of portClass = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN.
If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

20. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of portClass = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

21. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used s by the ONNS application.

Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

22. **nemspm** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).
23. **networkTNAType** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP
24. **networkTNAAddress** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter in case explicit setting of the default value is allowed.

25. **`outql`** – Outgoing timing quality

This parameter provides the outgoing quality level message on the S1 SDH MS overhead byte.

Type: Alphanumeric value set.

Range:

- PRC
- SSUL
- SSUT
- SEC
- DNU (default).

26. **`pmmsfeses`** – Performance Monitoring MS Far-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default: for STM1: 28800.
27. **`pmmsneses`** – Performance Monitoring MS Near-End SES Threshold
This parameter describes the Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: STM1: 1 – 192000.
Default: for STM1: 28800.
28. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.
29. **`pmrsneses`** – Performance Monitoring RS Near-End SES Threshold
This parameter describes the RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) per second to declare a second as severly errored.
Type: numeric value set.
Range: 1 – 64000.
Default: for STM1: 2400.
30. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

31. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

32. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

33. **`ptype`** – Port Type
This identifies the port type of the AID.
Type: Alphanumeric value set.
Range: STM1E.

34. **`rsmstca`** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.

35. **`rspm`** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

36. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

37. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
38. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
39. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.
40. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
41. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.

This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

42. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

43. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc).
Type: Numeric string.
Default: 2

44. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

45. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.

46. **strcwfmt** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout).
Type: Numeric string.
Default: 2
47. **tcamd** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).
48. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED.
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
49. **unequiposig** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-STM1E** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-STM1E** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-STM1E:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   "1-1-#-#-1-1:ptype=STM1E,frcdus=DISABLE,dexcthr=-3,degthr=-6,
      pmode=MON,...,pmmsneses=15,pmrsneses=30"
```

;

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-STM1E command.

Check CCB against ptype and istd

See definition of error response in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899).

Related TL1 messages

“ED-STM1E: Edit STM1E port” (p. 3-870)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STM1: Retrieve STM1 port

The command **RTRV-STM1** is used to retrieve all parameters related to the optical STM1 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STM1**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an STM1 port.
   Type: AID.
   Range: SPortAID of STM1 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-STM1** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Access Identifier of STM1 port.
   Type: AID.
   Range: SPortAID of STM1 port.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter inputsig showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-… N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **`femspm`** – Far-end Multiplex Section PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.

Range:

- ENABLE , set a forced DUS/DNU
- DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output *aid*.
A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
This parameter is a list of VCn rates across the port's bandwidth.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.
13. **`istd`** – Interface Standard
Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
Type: Alphanumeric value set.
Range: SDH.
Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
14. **`linkResCls`** – Link Resource Class
The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 or VC4 (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default) or VC4.

20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of portClass = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN.
If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of portClass = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of portClass = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string
Range: Each numeric string: 0…255
Default: 0-0-0-0

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
    The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: IPv4 format: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
26. **`nemspm`** – MS Near-End PM Enable
    Enable or disable near-end multiplex section (near-end line) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
27. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: Alphanumeric value set.
    Range:
    - IPV4 (default)
    - IPV6
    - NSAP
28. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **`opdis`** – Optical Distance
It shows the type of application for which the optical transmitter on this port is suited.
Type: Alphanumeric value set.
Range:

- INTRA-OFFICE
- SHORT-HAUL
- LONG-HAUL
- VERY-LONG-HAUL
- OLS_PASSIVE
- OLS_400G

- OLS_800G
- XTREME
- NA

30. **`opwl`** – Optical Wavelength
 Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
 Type: Alphanumeric value set.
 Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM), NA (for client ports). T
 he value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.
31. **`outql`** – Outgoing timing quality
 This parameter provides the outgoing quality level message on the S1 SDH MS overhead byte.
 Type: Alphanumeric value set.
 Range:
 - PRC
 - SSUL
 - SSUT
 - SEC
 - DNU (default).
32. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
 This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
 Type: numeric value set.
 Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
 - STM-1/OC-3: 1 – 192000.

 Default: The default values differ for SDH and SONET:
 - STM1: 28800
 - OC3: 154.

33. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
    This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

    - STM-1/OC-3: 1 – 192000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 28800
    - OC3: 154.

34. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.

    Range:

    - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
    - MON indicates that alarm monitoring is taking place on the port.
    - NMON means that alarm monitoring is not occurring on the port.

35. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
    This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of allowed ranges for SDH and SONET):1 – 64000.

    Default: The default values differ for SDH and SONET and the different rates:

    - STM1: 2400
    - OC3: 155.

36. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

37. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT

38. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …25. The unit of this value is ″milliseconds″.

39. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: STM1.

40. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

41. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295

42. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
43. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
44. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
45. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
46. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.

Range:

- ENABLE (default): enable consequent action for TIM
- DISABLE: disable consequent action for TIM.

47. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
48. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.
49. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.
50. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
51. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

53. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
   - 16: for 16 byte specific string (ITU/ETSI mode 1).
   - 64: for 64 byte string (SONET).

54. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.

Range:

- 1 byte string
- 16 bytes string
- 64 bytes string.

Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

55. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
56. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
57. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
Range: TR or TRRTR (default).

58. **trbmd** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED.
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.

59. **unequiposig** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-STM1** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-STM1** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-STM1:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=STM1,frcdus=DISABLE,dexcthr=-3,degthr=-6,
      pmode=MON,...,pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-STM1 command.

Check CCB against ptype and istd

See definition of error response in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899).

Related TL1 messages

"ED-STM1: Edit STM1 port" (p. 3-915)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STM4: Retrieve STM4 port

The command **RTRV-STM4** is used to retrieve all parameters related to the optical STM4 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STM4**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an STM4 port.
   Type: AID.
   Range: SPortAID of STM4 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-STM4** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
      ...
      ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Access Identifier of STM-4 port.
   Type: AID.
   Range: SPortAID of STM-4 port.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
   This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
   Type: numeric value set.
   Range: NX-NX-NX-…
   N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.
6. **`degthr`** – Signal Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
7. **`dexcthr`** – Excessive Degrade Threshold
   Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
   Type: numeric value set.
   Range (as exponents of 10): –5; –6; –7; –8; –9.
   Default: –6.
8. **`exprstrc`** – Value of Expected J0
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
9. **`femspm`** – Far-end Multiplex Section PM Enable
   Enable or disable of the performance monitoring of the multiplex section.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.

Range:

- ENABLE , set a forced DUS/DNU
- DISABLE (default), clear a forced DUS/DNU

11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).

12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4 or 4C representing VC3, VC4 or VC4-4C.
    Default: 44.

13. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).

14. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
16. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type: numeric value set.
    Range: 0 (default) to 4294967295
    The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
17. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
    The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3, VC4 or VC44C (default)
19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
    The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range: VC3 (default), VC4 or VC44C
20. **`ndetmd`** – Neighbor Detection Mode
    The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set.
    Range:
    AUTO, MAN
    Default:
    For *portClass* = INNI: AUTO
    For *portClass* = ENNI: MAN
21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
    This Parameter identifies the unique node identifier of the neighbor node of ports of portClass = INNI to be used for ONNS application. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN.
    If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.

neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **neighborSigPCIPAddr** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **nemspm** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **networkTNAType** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAType", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **`networkTNAAddress`** – Network TNA Address
The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type:
   - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
   - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
   - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

   Range:
   - IPV4: Each numeric string: 0…255
   - IPV6: Each hexadecimal number: 0000…FFFF
   - NSAP: Hexadecimal number:
     0000000000000000000000000000000000000000…
     FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

   Examples:
   - IPV4: 129.12.3.178
   - IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
   - NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

   Default:
   - IPV4: 0.0.0.0
   - IPV6: 0-0-0-0-0-0-0-0
   - NSAP: 0000000000000000000000000000000000000000

   The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
29. **`opdis`** – Optical Distance
It shows the type of application for which the optical transmitter on this port is suited.
Type: Alphanumeric value set.

Range:

- INTRA-OFFICE
- SHORT-HAUL
- LONG-HAUL
- VERY-LONG-HAUL
- OLS_PASSIVE
- OLS_400G
- OLS_800G
- XTREME
- NA.

30. **`opwl`** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 471 … 1611 (for CWDM), NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

31. **`outql`** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SDH MS overhead byte.
Type: Alphanumeric value set.
Range:
    - PRC
    - SSUL
    - SSUT
    - SEC
    - DNU (default).

32. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-4/OC-12: 1 – 768000.

Default: The default values differ for SDH and SONET:

- STM4: 115200
- OC12: 615.

33. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
   - STM-4/OC-12: 1 – 768000.

   Default: The default values differ for SDH and SONET and the different rates:
   - STM4: 115200
   - OC3: 615.
34. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.
Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
35. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000.
Default: The default values differ for SDH and SONET and the different rates:
   - STM4: 2400
   - OC12: 616.

36. **`portClass`** – Port Class
    This parameter identifies the port class of the AID.
    In which role the port is used by the ONNS application is specified by this parameter *portClass*.
    Type: Alphanumeric value set.
    Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
    If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
37. **`portControl`** – Port Control (PortAdministrative Mode)
    The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
    Type: Alphanumeric value set.
    Range: NORMAL (default), LOCKOUT
38. **`propagationDelay`** – Propagation Delay
    This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
    propagationDelay is used by the ONNS application.
    Type: Numeric string.
    Range: 0(default) …25. The unit of this value is ″milliseconds″.
39. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: STM4.
40. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition: The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0

41. **remLinkID** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295
42. **remRAID** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0
43. **remTID** – Remote TID
The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the “RTRV-HDR: Retrieve header” (p. 3-2417) description.
Default: empty string: ″″
44. **rsmstca** – RS/MS TCA Profile Pointer
Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
Type: Alphanumeric string.
Range: String with maximum of 24 characters.
Default: DEFAULT0.
45. **rspm** – RS PM Enable
Enable or disable regenerator section (section) performance monitoring.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

46. **`rstimca`** – RS TIM Consequent Action Enable
Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.

47. **`rstimmon`** – RS TIM Monitor Enable
Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
Type: Alphanumeric value set.
Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

48. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET ASAP for the SONET ports of corresponding line rate (see definition of output parameter SONET in RTRV-OC12) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

49. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.

50. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.

Range:

- INSERVICE (default): Provisioning values for service are accepted
- OUTOFSERVICE: Provisioning values for service are not accepted.

51. **srlgList** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **strc** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

53. **strcfmt** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
   - 1: for specific repeating byte (SONET)
   - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)

- 16: for 16 byte specific string (ITU/ETSI mode 1).
- 64: for 64 byte string (SONET).

54. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
55. **`strcst`** – Status of incoming J0
This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
Type: Alphanumeric value set.
Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
56. **`strcwfmt`** – Format of outgoing J0
The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
57. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (lso known as the implicit reset method)
- TR-RTR (lso known as the explicit reset method).

Type: Alphanumeric value set.
Range: TR or TRRTR (default).

58. **`trbmd`** – Tributary mode
If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
Type: Alphanumeric value set.
Range: ADAPTIVE, FIXED.
Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
59. **`unequiposig`** – Unequipped Output Signal List
This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3 or 4 representing VC3 or VC4.
Default: 14.

Detailed behavior description

The command retrieves the parameters as described above.

The command **`RTRV-STM4`** does not generate a **`REPT DBCHG`** message.

Example input/output

The following example shows the successful execution of a **`RTRV-STM4`** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-STM4:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=STM4,frcdus=DISABLE,dexcthr=-3,degthr=-6,
      pmode=MON,...,pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-STM4 command.

Check CCB against ptype and istd

See definition of error response in "RTRV-STM16: Retrieve STM16 port" (p. 3-2899).

Related TL1 messages

"ED-STM4: Edit STM4 port" (p. 3-975)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STM16: Retrieve STM16 port

The command **RTRV-STM16** is used to retrieve all parameters related to the optical STM16 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STM16**:*tid:aid:ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Description: Access Identifier of an STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-STM16** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **`all`** is not allowed.

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems

reasonable for a possible extension of this attribute to other port classes in future. This parameter is supported by ports of *portClass* UNI.
Type: IPv4 format: Four dot separated numeric string.
Range: Each numeric string: 0…255
Default: 0.0.0.0

5. **`crstat`** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, or 4 representing VC3 or VC4.

6. **`degthr`** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.

7. **`dexcthr`** – Excessive Degrade Threshold
Retrieving of the excessive degrade/signal failure threshold in exponent of 10. This parameter is available when *sdsfmode* = POISSON
Type: numeric value set.
Range (as exponent of 10): –3; –4; –5.
Default: –3.

8. **`exprstrc`** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex

9. **`femspm`** – Far-end Multiplex Section PM Enable
Enable or disable of the performance monitoring of the multiplex section.
Type: Alphanumeric value set.
Range:
   - ENABLE
   - DISABLE (default).

10. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.
    Range:
    - ENABLE , set a forced DUS/DNU
    - DISABLE (default), clear a forced DUS/DNU
11. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
12. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
    Default: The initial value of X is 4.
    Example:
    An STM-16 signal might be represented by 93-14-14C-243 which implies 9 VC3s, 1 VC4, 1 VC4-4C, 24 VC3s. The representation is not unique, for example it could also be 33-63-14-14C-243. Worst case of string length for STM16: 47 characters.
13. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
14. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set (4 bytes), every bit defining one link class
Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
Default: 00 for each alphanumeric character

15. **`localLinkID`** – Local Link IDentity
The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

16. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkLPID”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of “no localLPID provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

17. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.

Range:

- YES
- NO (default).

18. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C or VC416C (default)

19. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C or VC416C

20. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

21. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This

parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If portClass = INNI, the default value shall have the meaning of ″no neighbor discovered″.

22. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of portClass = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if ndetmd=MAN. If ndetmd=MAN, this parameter shall indicate the value set by the operator. If ndetmd=AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

23. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if ndetmd=MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

24. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

25. **`neighborSigPCIPAddr`** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

26. **`nemspm`** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

27. **`networkTNAType`** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

28. **networkTNAAddress** – Network TNA Address

The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.

The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This attribute replaces attribute "uniNetworkTNAAddress", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.

This parameter is supported by ports of *portClass* UNI.

This parameter is supported by ports of *portClass* EDGE.

Type:

- networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
- networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
- networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

Range:

- IPV4: Each numeric string: 0…255
- IPV6: Each hexadecimal number: 0000…FFFF
- NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

Examples:

- IPV4: 129.12.3.178
- IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
- NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

Default:

- IPV4: 0.0.0.0
- IPV6: 0-0-0-0-0-0-0-0
- NSAP: 0000000000000000000000000000000000000000

The default value shall have the meaning of "no TNA Address provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

29. **opdis** – Optical Distance

It shows the type of application for which the optical transmitter on this port is suited.

Type: Alphanumeric value set.

Range:

- INTRA-OFFICE
- SHORT-HAUL
- LONG-HAUL
- VERY-LONG-HAUL
- OLS_PASSIVE
- OLS_400G
- OLS_800G
- XTREME
- NA.

30. **`opwl`** – Optical Wavelength
Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
Type: Alphanumeric value set.
Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM), NA (for client ports).
The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.
31. **`outql`** – Outgoing timing quality
This parameter provides the outgoing quality level message on the S1 SDH MS overhead byte.
Type: Alphanumeric value set.
Range:
    - PRC
    - SSUL
    - SSUT
    - SEC
    - DNU (default).
32. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

33. **`pmmsneses`** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

34. **`pmode`** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

35. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000
Default: The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
    - SONET OC192: 8554
    - SONET OC768: 22776.
36. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.
37. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT
38. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

39. **`ptype`** – Port Type
This identifies the port type of the AID.
Type: Alphanumeric value set.
Range: STM16.

40. **`remAdvNodeID`** – Remote Advertised Node IDentity
The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

41. **`remLinkID`** – Remote Link IDentity
The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

42. **`remRAID`** – Remote Routing Area IDentity
The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

43. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: TID or empty-string
    Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
    Default: empty string: ""
44. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
45. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
46. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
47. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.

Range:

- ENABLE: enable TIM Monitor
- DISABLE (default): disable TIM Monitor

48. **`sdhpn`** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC48: Retrieve OC48 port" (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

49. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in Burst or Poisson mode.
Type: Alphanumeric value set.
Range:
    - BURST (default)
    - POISSON.

50. **`servcond`** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

51. **`srlgList`** – Shared Risk Link Group Identities List
The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Ten numeric strings separated by dashes.
Range: Each numeric string: 0…65535.
Default: 0-0-0-0-0-0-0-0-0-0.

52. **`strc`** – Value of incoming J0
    This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
    Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
    Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).
53. **`strcfmt`** – Format of incoming J0
    The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
54. **`strcout`** – Value of outgoing J0
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex

55. **`strcst`** – Status of incoming J0
    This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
    Type: Alphanumeric value set.
    Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
56. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.

    Range:

    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
57. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.

    Two thresholding methods are applicable:

    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).
58. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
    Type: Alphanumeric value set.
    Range: ADAPTIVE, FIXED.
    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
59. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the "unequipped" signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for tx and rx side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C or 16C representing VC3, VC4, VC4-4C or VC4-16C.
    Default: 164 = all VC4.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-STM16** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-STM16** command. Not all output parameters are listed (missing ones indicated by …):

```
 RTRV-STM16:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=STM16,frcdus=DISABLE,dexcthr=-3,degthr=-
      6,pmode=MON,...,pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-STM16 command.

Validation rule: Check CCB against *ptype* and *istd*

Check wether the CCB fits to *ptype* and *istd*.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=STM16 is made on a OP2G5 port which is provisioned as *istd* = SONET and *ptype* = OC48.

Related TL1 messages

"ED-STM16: Edit STM16 port" (p. 3-1035)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STM16T: Retrieve STM16T port

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **`RTRV-STM16T`**:*tid*:*aid*:*ctag*;

Input parameters

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   Description: Access Identifier of an STM16T port.
   Type: AID.
   Range: SPortAID of STM16T port.
   The keyword **`all`** is allowed up to shelf level.
3. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-STM16 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Access Identifier of STM16 port.
   Type: AID.
   Range: SPortAID of STM16 port.
   The keyword **`all`** is not allowed.

Spec block parameters

1. **`exprstrc`** – Value of Expected J0
   For RTRV-STM16T this parametr refers to the incoming (inress) direction
   Expected regenerator section trace identifier.
   The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
2. **`exprstrcout`** – Value of Expected J0 in outgoing direction
   Expected regenerator section trace identifier in outgoing (egress) direction.
   The parameter expstrcout specifying the expected path trace is only relevant for the section trace supervision in outgoing direction when the related trace read format strcfmtout is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex
3. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
   This parameter specifies the current far-side facility loopback status of the considered output *aid*.
   A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
   Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
   Type: Alphanumeric value set.

   Range:

   - YES
   - NO (default).

4. **`istd`** – Interface Standard
   Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
   Type: Alphanumeric value set.
   Range: SDH.
   Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
5. **`lpbk`** – Loopback
   This parameter specifies the current near-side loopback status of the considered output *aid*.
   A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
   Type: Alphanumeric value set.
   Range:
   - YES
   - NO (default).
6. **`opdis`** – Optical Distance
   It shows the type of application for which the optical transmitter on this port is suited.
   Type: Alphanumeric value set.
   Range:
   - INTRA-OFFICE
   - SHORT-HAUL
   - LONG-HAUL
   - VERY-LONG-HAUL
   - OLS_PASSIVE
   - OLS_400G
   - OLS_800G
   - XTREME.
7. **`opwl`** – Optical Wavelength
   Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
   Type: Alphanumeric value set.
   Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM), NA (for client ports).
   The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

8. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.
   Range:
   - AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.
   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.
9. **`pmrsneses`** – Performance Monitoring Section/RS Near-End SES Threshold
   This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
   Type: numeric value set.
   Range (superset of allowed ranges for SDH and SONET):1 – 64000
   Default: The default values differ for SDH and SONET and the different rates:
   - SDH (all rates): 2400
   - SONET OC48: 2392
   - SONET OC192: 8554
   - SONET OC768: 22776.
10. **`pmrsnesesout`** – Performance Monitoring Section/RS Near-End SES Threshold in outgoing direction
    This parameter describes the Section/RS Near-End SES threshold value for the outgoing (egress) direction. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET.
    Type: numeric value set
    Range (superset of allowed ranges for SDH and SONET): 1 – 64000
    Default:
    The default values differ for SDH and SONET and the different rates:
    - SDH (all rates): 2400
    - SONET OC48: 2392
11. **`ptype`** – Port Type
    This identifies the port type of the AID.
    Type: Alphanumeric value set.
    Range: STM16T.

12. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
13. **`rspm`** – RS PM Enable
    For RTRV-STM16T this parameter refers to the incoming (ingress) direction.
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.

    Range:

    - ENABLE
    - DISABLE (default).
14. **`rspmout`** – RS PM Enable in Outgoing Direction
    Enable or disable regenerator section (section) performance monitoring for the outgoing (egress) direction.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
15. **`rstimmon`** – RS TIM Monitor Enable
    For RTRV-STM16T this parameter refers to the incoming (Ingress) direction.
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.

    Range:

    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor
16. **`rstimmonout`** – RS TIM Monitor Enable in Outgoing direction
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0) in outgoing (egress) direction. This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode).
    Type: Alphanumeric value set.
    Range: ENABLE – enable TIM Monitor, DISABLE (default) – disable TIM Monitor

17. **sdhpn** – SDH ASAP
This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in “RTRV-OC48: Retrieve OC48 port” (p. 3-2638)) to be compatible with current generation applications.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters.
Default: DEFAULT.

18. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.

19. **strc** – Value of incoming J0
For RTRV-STM16T this parameter refers to the incoming (Ingress) direction.
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port.
The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

20. **strcfmt** – Format of incoming J0
For RTRV-STM16T this parameter refers to the incoming (Ingress) direction.
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

21. **`strcfmtout`** – Format of outgoing J0
    The read format of J0 in outgoing (egress) direction, the outgoing section trace (strcout) and the expected section trace (exprstrcout).
    Type: Numeric string.
    Range:
    - 1, for specific repeating byte (SONET)
    - 2 (default), for non-specific repeating byte (ITU/ETSI mode 2)
    - 16, for 16 byte specific string (ITU/ETSI mode 1)
    - 64, for 64 byte string (SONET).
22. **`strcout`** – Value of outgoing J0
    For RTRV-STM16T this parameter refers to the outgoing (egress) direction.
    This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
    If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
    Type: Hexadecimal digits.
    Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.
    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
    Default: 01Hex
23. **`strcst`** – Status of incoming J0
    For RTRV-STM16T this parameter refers to the incoming (ingress) direction.
    This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
    Type: Alphanumeric value set.
    Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
24. **`strcstout`** – Status of outgoing J0
    This parameter shows the status of the value of the RS/Section Trace Identifier in outgoing (egress) direction which is available with parameter *strcout*.
    alphanumeric value set

Range:

- NORMAL, *strcout* contains a valid value
- UNAVAILABLE, no value for *strcout* could be obtained by the system
- AIS, no value for *strcout* due to an SF condition

25. **`tcamd`** – TCA mode

This parameter shows the modus of the PM thresholding.

Two thresholding methods are applicable:

- TR-only (lso known as the implicit reset method)
- TR-RTR (lso known as the explicit reset method).

Type: Alphanumeric value set.
Range: TR or TRRTR (default).

Detailed behavior description

The command retrieves the parameters as described above.

The **`RTRV-STM16T`** command does not generate a **`REPT DBCHG`** message.

Example input/output

The following example shows the successful execution of a **`RTRV-STM16T`** command. Not all output parameters are listed (missing ones indicated by …):

```
 RTRV-STM16T:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=STM16T,pmode=MON,...,pmrsneses=30,pmrsfeses=30″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **`RTRV-STM16T`** command.

Validation rule: Check CCB against *ptype* and *istd*

Check wether the CCB fits to *ptype* and *istd*.

Error code:

`ICNV – Input, Command Not Valid, Interface Standard does not match`

E.g. DENY, if a command with rr=STM16T is made on a OPT2G5 port which is provisioned as *istd* = SONET and *ptype* = OC48T.

Related TL1 messages

"ED-STM16T: Edit STM16T port" (p. 3-1097)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STM64: Retrieve STM64 port

The command **RTRV-STM64** is used to retrieve all parameters related to the optical STM64 port or the STM64 client .

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STM64**:*tid:aid:ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   Access Identifier of an STM64 port.
   Type: AID.
   Range: SPortAID of ST-64 port.
   The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid **RTRV-STM64** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command Parameter

1. **`aid`** – Access Identifier
   Type: AID
   Range: STM-64 port AID

Spec block parameters

1. **`adminCost`**- Administrative Cost (Link Cost)
   The *adminCost* identifies the provisioned administrative link cost setting for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This parameter is supported by ports of *portClass* INNI.
   This parameter is supported by ports of *portClass* ENNI.
   Type: Numeric string.
   Range: 1(default)…65535
2. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
3. **`brstthr`** – Bursty Error Threshold
   Degradation threshold in a range from 5% to 50% with a resolution of 5% steps. This parameter applies when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, … 50.
   Default: 30.
4. **`clientNodeID`** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other port classes in future.
   This parameter is supported by ports of *portClass* UNI.
   Type: IPv4 format: Four dot separated numeric string.
   Range: Each numeric string: 0…255
   Default: 0.0.0.0

5. **crstat** – Constituent Rates Status list
This parameter shows the rates of the port signals. For fixed mode (*trbmd* = FIXED) this parameter is derived from the parameter *inputsig* showing the Tributary Input Rate Signal List.
Type: numeric value set.
Range: NX-NX-NX-…
N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC-64C.
6. **degthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
7. **dexcthr** – Excessive Degrade Threshold
Signal degrade threshold in a range from 10exp-5 to 10exp-9. This parameter applies when *sdsfmode* = POISSON.
Type: numeric value set.
Range (as exponents of 10): –5; –6; –7; –8; –9.
Default: –6.
8. **exprstrc** – Value of Expected J0
Expected regenerator section trace identifier.
The parameter *expstrc* specifying the expected path trace is only relevant for the section trace supervision when the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and *ndetmd* is set to AUTO, the parameter *exprstrc* is autonomously set by the neighbor detection mechanism.
Type: Hexadecimal digits
Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 89hex followed by 15 bytes 00hex
9. **fec** – FEC Enable
Forward Error Correction Enable. Enable to include the FEC information in the outgoing signal.
Type: Alphanumeric value set.
Range:
DISABLE (only) for:
   - OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
   - OC192 / STM64 intermediate reach / SH passive WDM compatible optics
   - OC192 / STM64 Long Reach / Long Haul 1550 nm optics
   - OC192 / STM64 low cost intra office 1310 nm optics

- OC192 / STM64 client ports (related to OTN ports)
- OC768 / STM256 basis low cost intra office 1310 nm optics.

ENABLE / DISABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

ENABLE (only) for:

- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G.

Default:

DISABLE for:

- OC192 / STM64 Intermediate Reach / Short Haul 1550 nm optics
- OC192 / STM64 intermediate reach / SH 32 colour passive WDM compatible optics
- OC192 / STM64 Long Reach / Long Haul 1550 nm optics
- OC192 / STM64 low cost intra office 1310 nm optics
- OC192 / STM64 client ports (related to OTN ports)
- OC768 / STM256 basis low cost intra office 1310 nm optics.

ENABLE for:

- OC192 / STM64 OLS800G compatible 80 colour optics
- OC192 / STM64 OLS Xtreme compatible 128 colour optics 40G
- OC768 / STM256 Long Reach / Long Haul 1550 nm optics (80km)
- OC768 / STM256 OLS Xtreme compatible 64 colour optics (extended L band)

10. **`femspm`** – Far-end Multiplex Section PM Enable
    Enable or disable of the performance monitoring of the multiplex section.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
11. **`frcdus`** – Force to DUS
    Force DUS/DNU for sync messaging on optical interfaces. When provisioned for ENABLE, the message ″do not use″ will be transmitted from that optical interface.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: (set a forced DUS/DNU)
    - DISABLE (default): (clear a forced DUS/DNU).

12. **`fsflpbkstat`** – Far Side Facililiy Loopback Status
    This parameter specifies the current far-side facility loopback status of the considered output *aid*.
    A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback existing for the aid.
    Note: not both loopbacks (*lpbk* and *fsflpbkstat*) can be set to YES at the same time.
    Type: Alphanumeric value set.

    Range:

    - YES
    - NO (default).
13. **`inputsig`** – Tributary Input Signal Rate List
    This parameter is a list of VCn rates across the port's bandwidth.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
    Default: 644.
14. **`istd`** – Interface Standard
    Each synchronous optical interface can be set to SDH or SONET. The default value depends on the setting of the parameter istddflt.
    Type: Alphanumeric value set.
    Range: SDH.
    Default: As specified via *istddflt* on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400).
15. **`linkResCls`** – Link Resource Class
    The *linkResourceClass* provides a classification for routing purposes of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it. The value of this attribute represents a 32-bit mask.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Alphanumeric value set (4 bytes), every bit defining one link class
    Range: fixed 4 alphanumeric characters with hexadecimal values 00..FF
    Default: 00 for each alphanumeric character
16. **`localLinkID`** – Local Link IDentity
    The *localLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be

retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: numeric value set
Range: 0 (default) to 4294967295

17. **`localLPID`** – Local Logical Port IDentity
The *localLPID* provides a unique logical identification of a single port with a *portClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
Depending on the *portClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute "uniNetworkLPID", as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: numeric value set.
Range: 0 (default) to 4294967295
The default value shall have the meaning of "no localLPID provisioned" for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.

18. **`lpbk`** – Loopback
This parameter specifies the current near-side loopback status of the considered output *aid*.
A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
Type: Alphanumeric value set.
Range:
    - YES
    - NO (default).

19. **`maxNNXCRate`** – Maximum NN Cross Connect Rate
The *maxNNXCRate* identifies the maximum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3, VC4, VC44C, VC416C or VC464C (default)

20. **`minNNXCRate`** – Minimum NN Cross Connect Rate
The *minNNXCRate* identifies the minimum rate at which cross-connections are allowed for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range: VC3 (default), VC4, VC44C, VC416C or VC464C

21. **`msrei`** – MS-REI mode
An special extended MS REI mode is standardized for STM64/OC192 interfaces. To support the interworking with older STM64/OC192 interfaces the single byte mode (non-extended) should also be supported.
Type: Alphanumeric value set.
Range: SINGLE or EXTENDED (default)

22. **`ndetmd`** – Neighbor Detection Mode
The *ndetmd* identifies the mode for neighbor detection for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* INNI.
This parameter is supported by ports of *portClass* ENNI.
Type: Alphanumeric value set.
Range:
AUTO, MAN
Default:
For *portClass* = INNI: AUTO
For *portClass* = ENNI: MAN

23. **`neighborASTNNodeID`** – Neighbor ASTN Node Id
This parameter identifies the unique node identifier of the neighbor node of ports of *portClass* = INNI to be used for ONNS application. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic

neighbor detection.
neighborASTNNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0 If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″.

24. **`neighborNodeID`** – Neighbor Node Id
This parameter identifies the IP address of the neighbor node of ports of *portClass* = INNI. For all ports with other values of portClass, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until portClass is INNI. This parameter can only be modified, if *ndetmd* = MAN. If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborNodeID is used by the ONNS application.
Type: Four dot separated numeric string.
Range: Each numeric string: 0…255.
Default: 0.0.0.0
If *portClass* = INNI, the default value shall have the meaning of ″no neighbor discovered″

25. **`neighborPortID`** – Neighbor Port Id
This parameter identifies the identity of the adjacent port at the neighbor NE for ports of *portClass* = INNI. For all ports with other values of *portClass*, this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI. This parameter can only be modified, if *ndetmd* = MAN.
If *ndetmd* = MAN, this parameter shall indicate the value set by the operator. If *ndetmd* = AUTO, this parameter shall indicate the value detected via automatic neighbor detection.
neighborPortID is used by the ONNS application.
Type: Four dash separated numeric string.
Range: Each numeric string: 0…255.
Default: 0-0-0-0.
Interpretation: For UNITE ports the contents of this parameter is interpreted as SPort AID, a value of a-b-c-d is interpreted as the SPortAID a-b-#-#-c-d.

26. **`neighborSigPCID`** – Neighbor Signaling PC IDentity
The *neighborSigPCID* provides a logical identification of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be

evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

27. **neighborSigPCIPAddr** – Neighbor Signaling PC IP Address
The *neighborSigPCIPAddr* identifies the IP address of the neighbor node connected via the corresponding port for signaling purposes, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This parameter is supported by ports of *portClass* ENNI.
Type: IPv4 format: Four dot separated numeric string
Range: Each numeric string: 0…255
Default: 0.0.0.0

28. **nemspm** – MS Near-End PM Enable
Enable or disable near-end multiplex section (near-end line) performance monitoring.
Type: Alphanumeric value set.
Range:
    - ENABLE
    - DISABLE (default).

29. **networkTNAType** – Network TNA Type
The *networkTNAType* identifies the type of the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
This attribute replaces attribute “uniNetworkTNAType”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
This parameter is supported by ports of *portClass* UNI.
This parameter is supported by ports of *portClass* EDGE.
Type: Alphanumeric value set.
Range:
    - IPV4 (default)
    - IPV6
    - NSAP

30. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNAType*. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This attribute replaces attribute “uniNetworkTNAAddress”, as starting with R9 this attribute is not exclusively valid any more for port class UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *portClass* UNI.
    This parameter is supported by ports of *portClass* EDGE.
    Type:
    - networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    - networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    - networkTNAType= NSAP: hexadecimal number (represents 20 Bytes)

    Range:
    - IPV4: Each numeric string: 0…255
    - IPV6: Each hexadecimal number: 0000…FFFF
    - NSAP: Hexadecimal number: 0000000000000000000000000000000000000000… FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

    Examples:
    - IPV4: 129.12.3.178
    - IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    - NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF

    Default:
    - IPV4: 0.0.0.0
    - IPV6: 0-0-0-0-0-0-0-0
    - NSAP: 0000000000000000000000000000000000000000

    The default value shall have the meaning of “no TNA Address provisioned” for ports with *portClass* value supporting this parameter, in case explicit setting of the default value is allowed.
31. **`och`** – optical channel enable
    Enable or disable of the optical channel. If the optical channel is disabled the further optical channel parameters don’t have any impact on transmission.
    Type: Alphanumeric value set.

Range:

- for ports on OP10 packs, which support to enable or disable OCH: ENABLE: enable optical channel; DISABLE: disable optical channel
- for ports on OP10 and OP40 packs, which cannot support OCH: DISABLE: disable optical channel
- for ports on OP10 and OP40 packs, which only support OCH: ENABLE: enable optical channel
- for client ports on packs which support OTN: DISABLE disable, optical channel not applicable.

Default:

- for OP10 and OP40 packs, which cannot support OCH: DISABLE
- for OP10 packs, which cannot support OCH: ENABLE
- for OP10 and OP40 packs, which only support OCH: ENABLE
- for client ports on packs which support OTN: DISABLE.

32. **`opdis`** – Optical Distance
    It shows the type of application for which the optical transmitter on this port is suited.
    Type: Alphanumeric value set.
    Range:
    - INTRA-OFFICE
    - SHORT-HAUL
    - LONG-HAUL
    - VERY-LONG-HAUL
    - OLS_PASSIVE
    - OLS_400G
    - OLS_800G
    - XTREME.
33. **`opwl`** – Optical Wavelength
    Optical Wavelength in nanometers. It shows the type of optics that is used on this port.
    Type: Alphanumeric value set.
    Range (wavelength in [nm]): 0 (i.e. undefined), 1310; 1550; 1530.334 … 1562.640 (i.e. C-Band); 1553.731 … 1608.329 (i.e. L-Band), 1471 … 1611 (for CWDM), NA (for client ports).
    The value undefined is used e.g. in the context of optical modules if the retrieval of the inventory data is not yet supported.

34. **`otnclient`** – OTN Client Port
    This parameter indicates whether the port is an OTN Client Port or an optical SDH/SONET port..
    Type: Alphanumeric value set.
    Range: YES for OTN client port; NO for optical SDH/SONET port.
35. **`outql`** – Outgoing timing quality
    This parameter provides the outgoing quality level message on the S1 SDH MS overhead byte.
    Type: Alphanumeric value set.

    Range:
    - PRC
    - SSUL
    - SSUT
    - SEC
    - DNU (default).
36. **`pmmsfeses`** – Performance Monitoring Line/MS Far-End SES Threshold
    This parameter describes the Line/Multiplex-Section Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.

    Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):
    - STM-16/OC-48: 1 – 3072000
    - STM-64/OC-192: 1 – 12288000
    - STM-256/OC-768: 1 – 49152000.

    Default: The default values differ for SDH and SONET and the different rates:
    - STM16: 460800
    - STM64: 1843200
    - STM256: 7372800
    - SONET OC48: 2459
    - SONET OC192: 9835
    - SONET OC768: 39340.

37. **pmmsneses** – Performance Monitoring Line/MS Near-End SES Threshold
This parameter describes the Line/Multiplex-Section Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.

Range (the range depends on the line rate but is identical for the corresponding SDH and SONET rates):

- STM-16/OC-48: 1 – 3072000
- STM-64/OC-192: 1 – 12288000
- STM-256/OC-768: 1 – 49152000.

Default: The default values differ for SDH and SONET and the different rates:

- STM16: 460800
- STM64: 1843200
- STM256: 7372800
- SONET OC48: 2459
- SONET OC192: 9835
- SONET OC768: 39340.

38. **pmode** – Port Mode
This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
Type: Alphanumeric value set.

Range:

- AUTO (default) indicates that, when a signal is detected at the port, then alarm monitoring will start.

  The value AUTO is only supported for optical SDH/SONET ports, i.e. not supported for client ports.
- MON indicates that alarm monitoring is taking place on the port.
- NMON means that alarm monitoring is not occurring on the port.

Default: AUTO for optical SDH/SONET ports; NMON for client ports.

39. **pmrsneses** – Performance Monitoring Section/RS Near-End SES Threshold
This parameter describes the Section/RS Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of allowed ranges for SDH and SONET):1 – 64000

Default: The default values differ for SDH and SONET and the different rates:

- SDH (all rates): 2400
- SONET OC48: 2392
- SONET OC192: 8554
- SONET OC768: 22776.

40. **`polynomial`** – Optical Channel Scrambling Polynomial
This parameter is used by the Wavewrapper scrambler on the optical channel. This parameter is relevant only if the optical channel is enabled (*och* = ENABLE).
Type: numeric value set.
Range: 1x7 + x + 1; 2x16 + x12 + x3 + x + 1.
Default: 2.

41. **`portClass`** – Port Class
This parameter identifies the port class of the AID.
In which role the port is used by the ONNS application is specified by this parameter *portClass*.
Type: Alphanumeric value set.
Range: TRADITIONAL, INNI, EDGE, UNI, ENNI.
Default: If Cross Connect Setup (*xcsetup*) is set to SWISS, then the value of the system parameter Port Class Default (*portclassdefault*) is used as default value. Otherwise, the default value is TRADITIONAL.

42. **`portControl`** – Port Control (PortAdministrative Mode)
The *portControl* identifies the administrative mode for a port with a *portClass* value used by the ONNS application. For ports of *portClass* TRADITIONAL this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is different from TRADITIONAL.
Type: Alphanumeric value set.
Range: NORMAL (default), LOCKOUT

43. **`propagationDelay`** – Propagation Delay
This parameter identifies the provisioned propagation delay value for ports of *portClass* = INNI. For all ports with other values of *portClass* this parameter is not relevant, i.e. parameter can be retrieved and changed, but will internally not be evaluated until *portClass* is INNI.
propagationDelay is used by the ONNS application.
Type: Numeric string.
Range: 0(default) …25. The unit of this value is ″milliseconds″.

44. **`ptype`** – Port Type
Definition: This identifies the port type of the AID.
It is derived from the functional name and functional qualifier of a circuit pack. If the functional qualifier indicates a pack with 10GE_WANPHY interfaces the port type has the value 10GE_WANPHY, else STM64.
Type: Alphanumeric value set.
Range: STM64.

45. **`remAdvNodeID`** – Remote Advertised Node IDentity
    The *remoteAdvNodeID* provides a logical identification of the remote level 1 routing entity the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter The character of this remote routing entity depends on the level 1 routing model configured for the corresponding neighbor node.
    For all ports with *portClass* values not supporting this parameter and in case a level 1 routing model not supporting this parameter is configured, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until both these conditions are fulfilled.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
46. **`remLinkID`** – Remote Link IDentity
    The *remoteLinkID* provides a logical identification of the link the corresponding port is assigned to, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the remote NEs point-of-view. For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: numeric value set
    Range: 0 (default) to 4294967295
47. **`remRAID`** – Remote Routing Area IDentity
    The *remoteRAID* provides a logical identification of the remote routing area the corresponding port is connected to under the following condition:
    The port has a *portClass* value used by the ONNS application and supporting this specific parameter For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Four dot separated numeric string
    Range: Each numeric string: 0…255
    Default: 0.0.0.0
48. **`remTID`** – Remote TID
    The *remoteTID* provides the TID of the node connected via the corresponding port, if the port has a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.

This parameter is supported by ports of *portClass* ENNI.
Type: TID or empty-string
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: ""

49. **`rsmstca`** – RS/MS TCA Profile Pointer
    Definition of the active regenerator section/section and multiplex section/line threshold crossing alarm profile pointer.
    Type: Alphanumeric string.
    Range: String with maximum of 24 characters.
    Default: DEFAULT0.
50. **`rspm`** – RS PM Enable
    Enable or disable regenerator section (section) performance monitoring.
    Type: Alphanumeric value set.
    Range:
    - ENABLE
    - DISABLE (default).
51. **`rstimca`** – RS TIM Consequent Action Enable
    Regenerator Section Trace Identifier Mismatch consequent action enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimca* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE (default): enable consequent action for TIM
    - DISABLE: disable consequent action for TIM.
52. **`rstimmon`** – RS TIM Monitor Enable
    Regenerator Section Trace Identifier Mismatch Monitor enable (J0).
    This parameter is relevant only if the related trace read format *strcfmt* is set to 16 (16-byte-specific-string mode). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *rstimmon* is autonomously set by the neighbor detection mechanism.
    Type: Alphanumeric value set.
    Range:
    - ENABLE: enable TIM Monitor
    - DISABLE (default): disable TIM Monitor

53. **`sdhpn`** – SDH ASAP
    This parameter retrieves the ASAP profile name of type PORT to which the designated AID points. The value of this parameter is identical to the SONET Alarm Severity Assignment Profile for the SONET ports (see definition of output parameter SONET in "RTRV-OC192: Retrieve OC192 port" (p. 3-2664)) to be compatible with current generation applications.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.
54. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in Burst or Poisson mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON.
55. **`servcond`** – Service Condition
    This parameter indicates its status concerning service involvement.
    Type: Alphanumeric value set.
    Range:
    - INSERVICE (default): Provisioning values for service are accepted
    - OUTOFSERVICE: Provisioning values for service are not accepted.
56. **`srlgList`** – Shared Risk Link Group Identities List
    The *srlgList* identifies the provisioned shared risk group identities for a port with a *portClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *portClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *portClass* is set to a value supporting it.
    The max. number of identities that can be provisioned per port may be release dependent. The order of the identities is not significant, e.g. 3-7-1 is considered the same as 7-1-3.
    If less than the max. number of identities are provisioned, the remaining ones are set to the default value 0, e.g. 3-2-1 given yields 3-2-1-0-0-0-0-0-0-0, if the max. number that can be given is 10.
    This parameter is supported by ports of *portClass* INNI.
    This parameter is supported by ports of *portClass* ENNI.
    Type: Ten numeric strings separated by dashes.
    Range: Each numeric string: 0…65535.
    Default: 0-0-0-0-0-0-0-0-0-0.

57. **`strc`** – Value of incoming J0
This parameter shows the value of the Section Trace Identifier on the incoming section overhead J0 byte of a port. The length of the string depends on the value of the parameter *strcfmt*. The status of this value is contained in parameter *strcst*. If *strcst* indicates NORMAL, then strc contains a valid value. If *strcst* indicates UNAVAILABLE or AIS, then no incoming section trace identifier could be obtained by the system, e.g. due to a signal-fail condition, and the output parameter strc is omitted from the response.
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range: 1 byte string, 16 byte string, 64 byte string (dependent on the strcfmt parameter).

58. **`strcfmt`** – Format of incoming J0
The read format of J0, the incoming section trace (strc) and the expected section trace (exprstrc). If the parameter *portClass* is set to INNI and ndetmd is set to AUTO, the parameter *strcfmt* is autonomously set by the neighbor detection mechanism.
Type: Numeric string.
Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).

59. **`strcout`** – Value of outgoing J0
This identifies the section trace message to be transmitted. The length is set with the *strcwfmt* parameter.
If ITU/ETSI mode 2 is selected (strcwfmt = 2) the outgoing J0 is fixed to 01hex. Therefore the setting of this parameter has no impact. If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcout* according to the actually outgoing J0.
Type: Hexadecimal digits.
Range:
    - 1 byte string
    - 16 bytes string
    - 64 bytes string.

    Dependent on the strcwfmt parameter. Each byte is expressed as two hexadecimal digits (example value: 6D). Value constraint for 16 byte format: The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
Default: 01Hex

60. **`strcst`** – Status of incoming J0
    This parameter shows the status of the value of the incoming RS/Section Trace Identifier which is available with parameter *strc*.
    Type: Alphanumeric value set.
    Range: NORMAL: strc contains a valid value; UNAVAILABLE: no value for strc could be obtained by the system; AIS: no value for strc due to an SF condition.
61. **`strcwfmt`** – Format of outgoing J0
    The write format of J0, the outgoing section trace (strcout). If the parameter *portClass* is set to INNI, the system shall autonomously set the parameter *strcwfmt* according to the actually outgoing J0.
    Type: Numeric string.
    Range:
    - 1: for specific repeating byte (SONET)
    - 2 (default): for non-specific repeating byte (ITU/ETSI mode 2)
    - 16: for 16 byte specific string (ITU/ETSI mode 1).
    - 64: for 64 byte string (SONET).
62. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.
    Two thresholding methods are applicable:
    - TR-only (lso known as the implicit reset method)
    - TR-RTR (lso known as the explicit reset method).

    Type: Alphanumeric value set.
    Range: TR or TRRTR (default).
63. **`trbmd`** – Tributary mode
    If *trbmd* is FIXED, then *inputsig* constrains the cross-connection rates that can be entered for the port. If *trbmd* is ADAPTIVE, then input *inputsig* has no effect.
    Type: Alphanumeric value set.
    Range: ADAPTIVE, FIXED.
    Default: The initial value is defined as the *trbmddflt* parameter on system level, see definition in "RTRV-EQPT (system): Retrieve Equipment (system)" (p. 3-2400) command.
64. **`unequiposig`** – Unequipped Output Signal List
    This parameter is a list of VCn rates across the port's bandwidth. It determines the rate(s) of the *unequipped* signal that will be transmitted if there is no cross-connection. *unequiposig* and *inputsig* do not need the identical setting. Mixed rates for *tx* and *rx* side of the port are allowed.
    Type: numeric value set.
    Range: NX-NX-NX-…
    N is the number of Xs. X can be 3, 4, 4C, 16C or C64 representing VC3, VC4, VC4-4C, VC4-16C or VC4-64C.
    Default: 644 = all VC4.

Detailed behavior description

The command retrieves the parameters as described above.

The command **RTRV-STM64** does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-STM64** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-STM64:LT-UNITE:1-1-#-#-1-1:123456;

   LT-UNITE 00-11-24 08:00:00
M  123456 COMPLD
   ″1-1-#-#-1-1:ptype=STM64,frcdus=DISABLE,dexcthr=-3,degthr=-6,
      pmode=MON,...,pmmsneses=15,pmrsneses=30″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-STM64 command.

Check CCB against ptype and istd

See definition of error response in “RTRV-STM16: Retrieve STM16 port” (p. 3-2899).

Related TL1 messages

“ED-STM64: Edit STM64 port” (p. 3-1111)

References

GR-199-CORE (RTRV-*rr*)

RTRV-STS1: Retrieve STS1 tributary

General

The command **RTRV-STS1** is used to retrieve all parameters related to the STS1 tributary.

Note: If a tributary AID (Access IDentifier) is specifed in the RTRV-STS1 request for a slot provisioned with a GE1 pack (e.g. 1-1-#-#-1-1-1 is used), then an empty reply is returned instead of: `IENE – Input, Entity Not Exists`.

For GE1 the “RTRV-VCGTRIB: Retrieve virtual concatenation group tributary ” (p. 3-3109) command must be used.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-STS1**:*tid*:*aid*:*ctag*;

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **aid** – Access Identifier
   AID of an SDH tributary. The keyword **all** is allowed on slot, port and tributary level. The so called ″dynamic parameters″ is reported, if a **all** is used at the tributary level.
   The ″dynamic parameters″ are:
   - *ptrc*, *ptrcst*, *siglb*, *trailstate*.

   Type: AID
   Range: STS1NumAID.
3. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for the input parameter syntax and description of this parameter.

   Important! To retrieve the dynamic parameters (*ptrc*, *ptrcst*, *siglb*, *trailstate*) of all tributaries use the “RTRV-ALL: Retrieve all (tributary)” (p. 3-2228) command and set the parameter *dyn* to YES.

Output format

In response to a valid RTRV-STS1 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   AID of the SONET tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   If the STS-1 tributary is related to a EC1 port, then the parameter *sdsfmode* is fixed to POISSON. As a consequence, the parameter *brstintvl* will never be used by the system in this case.
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   If the STS-1 tributary is related to a EC1 port, then the parameter *sdsfmode* is fixed to POISSON. As a consequence, the parameter *brstthr* will never be used by the system in this case.
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.

3. **`crslpbkrate`** – Cross-connect loopback Rate
   It is independent from the current interface standard and therefore both SDH and SONET values are allowed.
   The parameter *crslpbkrate* is not reported if the parameter *crlspbkstat* indicates that no loopback is set.
   Type: Alphanumeric value set.
   Range:
   - VC3
   - VC4
   - VC44C
   - VC416C
   - VC464C
   - STS1
   - STS3
   - STS12
   - STS48
   - STS192
4. **`crslpbkstat`** – Cross-connect loopback status
   State of cross- connect loopback.
   Type: Alphanumeric value set.
   Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
5. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
6. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
7. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.

8. **feppm** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
9. **loca** – Ring A-node TID
   This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
   loca is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in "RTRV-HDR: Retrieve header" (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
10. **locz** – Ring Z-node TID
   This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
   locz is introduced specifically for use related to ONNS application.
   Type: TID or empty-string.
   Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
   Default: empty string: "". This is displayed as \"\" in output direction.
11. **hptimmon** – Path Trace Monitoring Enable (J1)
   Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).
12. **neppm** – Near-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE (default).

13. **`pjepm`** – Pointer Justification Event PM Enable
    It indicates whether this performance monitoring feature is enabled or disabled.
    Type: Alphanumeric value set.

    Range:

    - ENABLE
    - DISABLE (default).

14. **`pmhpfeses`** – Performance Monitoring Path Far-End SES Threshold
    This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.

15. **`pmhpneses`** – Performance Monitoring Path Near-End SES Threshold
    This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
    Type: numeric value set.
    Range (superset of ranges for SDH and SONET): 1-64000.
    Default: 2400.

16. **`pse`** – PDI-P Switching Enable
    The PDI-P Switching Enable parameter indicates if the PDI-P contributes to the automatic path protection switch criteria.
    Type: Alphanumeric value set.

    Range:

    - ENABLE
    - DISABLE (default).

17. **`ptca`** – Path TCA Profile Pointer
    Type: Alphanumeric string.
    Range: string with up to 24 characters.
    Default: DEFAULT0.

18. **`ptrc`** – Path Trace (J1)
    This parameter shows the value of the path trace identifier on the incoming path overhead J1 byte of a non-terminated tributary. The length depends on the value of the parameter *ptrcrfmt*. The status of this value is contained in parameter *ptrcst*. If *ptrcst* indicates NORMAL, then ptrc contains a valid value. If *ptrcst* indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to a signal-fail condition or intra-NE

communication problems, and the output parameter ptrc is omitted from the response. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).

Range:

- 1 byte for non-specific repeating byte
- 16 byte: These 16 byte shall consist of 15 byte information and the CRC7 byte
- 64 bytes: This is a pure byte stream without any restrictions.

19. **`ptrcrfmt`** – Path Trace Read Format (J1)
The read format of J1, the incoming path trace (ptrc).
Type: numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16 byte specific string.
- 64: for 64 byte string.
Default: 2.

20. **`ptrcst`** – Status of Incoming Path Trace (J1)
This parameter shows the status of the value of the incoming path trace identifier which is available with parameter ptrc. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
Type: Alphanumeric value set.

Range:

- NORMAL: ptrc contains a valid value
- UNAVAILABLE: no valid value for ptrc could be obtained by the system
- AIS: no valid value for ptrc due to an SF condition.

21. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
The degraded state is detected in BURST or POISSON mode.
Type: Alphanumeric value set.

Range:

- BURST
- POISSON (default).

If the tributary is related to a EC1 port, then the range is limited to: POISSON

22. **`siglb`** – Signal Label
This parameter shows the value of the signal label on the incoming path overhead C2 byte of a non-terminated VC tributary.
The parameter is not reported if the *trailstate* parameter indicates a fail condition, i.e. has the value AIS or UNAVAILABLE. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
Type: Alphanumeric value set.
Range: 00, 01, …, FE, FF; i.e. two hexadecimal digits.

23. **`substr`** – Substructured

    This parameter indicates if the tributary is substructured.

    Substructure for LO cross-connections:

    A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.

    The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).

    A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **`ED-SUBS`** command.

    Substructure for conversion cross-connections:

    A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.

    The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.

    A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.

    Type: Alphanumeric value set.

    Range:

    - YES: Automatically sub structured for LO cross-connections
    - MANSUB: Manually sub structured for LO cross-connections
    - CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
    - NO (default): Not sub structured, not used for conversion cross-connections

    If the STS-1 tributary is related to a EC1 port, then no LO cross-connections are allowed. As a consequence, the parameter *substr* can only have the values CNV and NO in this case.

24. **`tmonmd`** – Tributary Monitoring Mode

    This parameter indicates the mode of the tributary.

    Type: Alphanumeric value set.

    Range:

    - MON = alarm monitoring is taking place on the port.
    - NMON (default) = alarm monitoring is not occurring on the port.

25. **`trailstate`** – Trail State (C2)

    It is possible to read the accepted trail value status (C2) of every VC which is terminated/monitored in the system. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).

    Type: Alphanumeric value set.

Range:

- NORMAL: for accepted TSL which has valid payload type
- AIS: for SSF, VC-AIS, dTIM or dUNEQ
- UNAVAILABLE: for TSL which does not meet the acceptance criteria or no reporting is possible.

Default: UNAVAILABLE.

26. **`tribsonetpn`** – Tributary Alarm Severity Assignment Profile
    This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
    Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    Default: DEFAULT.

Detailed behavior description

Applicability for STS3c and concatenated signals

The command **`RTRV-STS1`** is also applicable to STS3c tributaries. Even though an STS3c signal is not a concatenated STS1 signal in a strict sense, from a pointer processor viewpoint it can be considered to be comprised of three STS1 signals which have the properties of a concatenated signal. Thus it is not necessary to have a separate RTRV command for STS3c. Similarly, RTRV-STS1 is also applicable to other ″concatenated″ signals.

Retrieval of concatenated tributaries

The rules as defined for RTRV-VC3 apply as well.

Behavior for non-existing entities

If the specified VC3/VC4 tributaries do not exist, the command shall return an empty response. E.g. if an attempt is made, to retrieve VC3/VC4 tributaries on a GbE circuit pack.

Example input/output

```
RTRV-STS1:LT-UNITE:1-1-#-#-3-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-3-1-1:crslpbkstat=NO,tmonmd=NMON,tribsonetpn=Default,
ptca=Default0,feppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-
      6,...,ptrcrfmt=1,dexcthr=-3,ptrc=00,ptrcst=NORMAL,sdsfmode=POISSON″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-STS1 command.

Check CCB against *istd* of related port

See definition of error response in “RTRV-VC3: Retrieve VC3 tributary” (p. 3-3069).

Related TL1 messages

“ED-STS1: Edit STS1 tributary” (p. 3-1177)

References

GR-199-CORE (RTRV-*rr*)

RTRV-SYNCN: Retrieve synchronization

The command displays the provisioning and operational information of the synchronization attributes as set by the **SET-SYNCN** command and after initial system start-up. It also displays the status of the external timing reference input and output ports.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-SYNCN**:*tid*:*[aid]*:*ctag*;

Input parameters

Command parameters

1. **tid** – target identifier
   See definition of *tid – target identifier* in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – access identifier
   This parameter is used to identify the timing related settings. The AID separates the parameter in system timing related parameter, output timing related parameters, parameters related to assigned timing references and parameters for external timing input and external timing output. The *all* AID retrieves all retrievable parameters for synchronization. If the AID is omitted, the outputs for the whole synchronization area are given.
   Type: AID
   Range:
   - SystemAID
   - Timing_Output_AID
   - External_timing_ref_AID
   - Line_timing_ref_AID
   - External_Timing_input_port_AID
   - External_Timing_output_port_AID
   - **all**.

   Default: **all**.
3. **ctag** – correlation tag
   See definition of *ctag – correlation tag* in “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

After receiving the **RTRV-SYNCN** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "aid:spec_block"
       ...
       ...
   "aid:spec_block"
;
```

Output parameters

Command parameters

1. **aid** – access identifier
   This parameter specifies the timing related settings.
   Type: AID
   Range:
   - SystemAID
   - Timing_Output_AID
   - External_timing_ref_AID
   - Line_timing_ref_AID
   - External_Timing_input_port_AID
   - External_Timing_output_port_AID

   Default: Not applicable.

Spec block parameters

1. System Timing
   a. **active_ref** – active timing reference status
      This parameter displays the active timing reference, which the system clock is currently using for locked mode of system timing.
      The value NONE is reported in the following cases:
      - no timing reference is assigned (or all references are failed or locked-out)
      - the system is in free-running mode
      Type: text string
      Range: External_timing_ref_AID, Line_timing_ref_AID; NONE
      Default: NONE.
   b. **clkmod** – clock mode
      This parameter specifies the provisioned operational mode of the system timing.
      In the clockmod = FREE-RUNNING mode the system clock process uses the internal oscillator as reference. In the clockmod = LOCKED mode the system clock process uses the input signal from the system.

Type: Alphanumeric value set.

Range: FREE-RUNNING, LOCKED.

Default: FREE-RUNNING.

c. **`clkmod_stat`** – system clock mode status

This parameter specifies the state of the system clock. The range is depending on the system synchronization characteristic.

NORMAL: This operational mode indicates either the NE derives timing from the internal oscillator when clockmod = FREE-RUNNING, or the system timing is locked to a timing reference signal when clockmod = LOCKED and the reference is derived from one of the timing input signals. Refer to *clkmod* parameter.

AUTONOMOUS-HOLDOVER: This state indicates that all references are failed and no higher priority request is present.

FORCED-HOLDOVER: This operational mode is caused by a Clock Backup Request.

FAST-START: This operational mode indicates for fast pull-in of the clock when recovering from holdover (for SONET only).

The values AUTONOMOUS-HOLDOVER, FORCED-HOLDOVER and FAST-START are only possible when clockmod = LOCKED.

Type: Alphanumeric value set.

Range:

SONET: NORMAL, AUTONOMOUS-HOLDOVER, FORCED-HOLDOVER, FAST-START.

SDH: NORMAL, AUTONOMOUS-HOLDOVER, FORCED-HOLDOVER.

Default: NORMAL

d. **`refsw_stat`** – reference switch status

The System Timing Reference Switch Request State is a state of the system timing.

The value FORCED means, the system enters the Forced Switch state upon the initiation and successful completion of the Forced Switch command. While in the Forced Switch state the system may not switch the active timing reference, neither automatically nor by means of the Forced Switch or Manual Switch command.

The value MANUAL means, the system enters the Manual Switch state upon the initiation and successful completion of the Manual Switch command. While in the Manual Switch state the system may switch the active timing reference automatically if required for protection switching.

The parameter NO-REQUEST indicates, this is the routine-operation quiet state in which no external command activities are occurring. The system enters this state upon initial start-up, and following a clear command, or the clearing of an automatic switch.

Type: Alphanumeric value set.

Default: NO-REQUEST.

e. **`stratum_level`** – stratum level status

This parameter provides information about the level of the system clock. The parameter is depending from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: 3

SDH: SEC

Default:

SONET: 3

SDH: SEC.

f. **`syncmsg`** – system synchronization status messaging mode

This parameter is used to ENABLE or DISABLE the SSM protocol.

The parameter provisions the automatic reference selection process to use the incoming QLs in QL-enabled mode or to ignore them in QL-disabled mode. In case of DISABLE, for SDH all output SSMs are set to DNU, for SONET all output SSMs are set to STU.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: ENABLE.

g. **`sysql_stat`** – system quality level status

This parameter indicates the overall system timing quality level status. If the system configured as SONET then the SONET related values are indicated and if the system configured as SDH the SDH related values are indicated.

Type: Alphanumeric value set.

Range:

SONET: PRS, STU, ST2, ST3, DUS

SDH: PRC, SSUL, SSUT, SEC, DNU.

Default:

SONET: DUS

SDH: DNU.

h. **`timing`** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

Default: Default

i. **`wtr`** – wait to restore

This parameter specifies the provisioned wait to restore timer used for all timing ports for revertive switching. This is the amount of time it takes to switch from lower priority to higher priority references. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: 0SEC, 20SEC, 0…60MIN (1 minute steps), 99 (infinite).

SDH: 0SEC, 0…60MIN (1 minute steps); the values “0SEC” and “0MIN” can be used for SONET and SDH to provision a zero wait to restore time.

Default:

SONET: 20SEC

SDH: 5MIN

j. **wu_stat** – warm up status

This parameter is an attribute of the timing pack hardware design.

At initial start up of the system and at replacing of the active or standby timing function the system is in warm-up state. DNU/DUS is transmitted during this state on all ports. The clock would be expected to warm up for a temporary period of time before transition to NORMAL.

Type: Alphanumeric value set.

Range: NORMAL, WARMING-UP.

Default: NORMAL.

2. Output Timing (instance only valid for SDH)

a. **active_oref_stat** – active output timing reference status

This parameter displays the current selected timing reference of the timing link switch for the external output.

The value NONE is reported in the following cases:

- the external output timing is not locked to system timing (i.e. extout_prov ≠ SETG) AND none of the six line_timing instances (line1 to line6) is enabled OR all are failed OR locked out

- the external output timing is locked to system timing (i.e. extout_prov = SETG)

Type: text string.

Range: Line_timing_ref_AID, NONE.

Default: NONE.

b. **orefswr_stat** – output timing reference switch request status

This parameter indicates the output timing reference switch request state.

The value MANUAL means, a manual switch is active to the external timing output. The value FORCED means, a forced switch is active to the external timing output. The value NO-REQUEST means, no external command activities are occurring at the external timing output.

Type: Alphanumeric value set.

Default: NO-REQUEST.

3. Assigned Timing Reference

a. **extout_pri** – timing output reference priority

This parameter specifies the priority assignment of the timing link switch of the external output timing. This parameter is only for SDH and for line timing references.

The greater the number the lower the priority. The number ″1″ indicates the highest priority. The priority ″0″ means not assigned for external output.

Type: Numeric value set.

Range: 0 (means DISABLED), 1, 2, 3, 4, 5, 6.

Default: 0 (means DISABLED).

b. **`inql`** – assigned timing reference quality value

This parameter displays the input synchronization message for each port of the timing link switch provisioning of the system timing and of the output timing. In the output timing context the values for SDH are valid only. In case of invalid SSM, the value DNU or DUS is displayed. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range: SONET: PRS, STU, ST2, ST3, DUS

SDH: PRC, SSUT, SSUL, SEC, DNU.

Default: Not applicable.

c. **`inql_stat`** – input quality status

This parameter indicates the synchronization messaging quality level status for the specified timing reference.

The parameter describes the validity of the incoming quality level and take the following values: VALID, when the QL is a valid SSM and stable, INVALID, when the QL is not a valid SSM, or unstable, NOT-SUPPORTED, when SSM reception is not supported on that interface due to HW capabilities or the chosen format of the input signal (such as 2MHz), NOT-APPLICABLE, when no port is assigned to the current timing reference or the assigned timing reference experiences a reference fail.

Type: Alphanumeric value set.

Range: VALID, INVALID, NOT-SUPPORTED, NOT-APPLICABLE.

Default: NOT-APPLICABLE.

d. **`ql_prov`** – quality provisioned

This parameter defines the quality for the specified timing reference to a fixed SSM value. It is the provisioned quality level sync message value that will be received on timing references.

This parameter is related to all assigned timing references (both external timing references and line timing references) and overrides what is received for the respective timing reference. The AUTO value implies sync message support. The AUTO value is rejected if SSM is not supported for the assigned timing reference (i.e. for DS1 SF). The value range depends from the system synchronization characteristic. Per default is:

AUTO – for STM-N/OC-N line, 2Mbit/s framed or DS1 ESF with SSM,

SSUT – for 2MHz, 2Mbit/s unframed or framed without SSM

STU – for DS1 SF or DS1 ESF without SSM

Type: Alphanumeric value set.

Range:

SONET: AUTO, PRS, STU, ST2, ST3

SDH: AUTO, PRC, SSUL, SSUT, SEC.

Default:

SONET: AUTO (External_timing_ref_AID, Line_timing_ref_AID)

SDH: AUTO (Line_timing_ref_AID) SSUT (External_timing_ref_AID).

e. **ref_lock** – timing reference lockout status

This parameter displays the timing reference lockout state to the specified timing reference. It is the input status for each port of the timing link switch provisioning of the system timing.

The value LOCKOUT means, the assigned reference has been locked out by the user, and cannot be used as a timing reference signal until the lockout is cleared.

Type: Alphanumeric value set.

Range: LOCKOUT, NO-LOCKOUT.

Default: NO-LOCKOUT.

f. **ref_stat** – assigned timing reference signal status

The Assigned Timing Reference Signal Status is a status of each timing reference.

The status NORMAL means, the timing reference is available and operating normally without faults. The status FAILED means, the timing reference is failed. The status WAIT-TO-RESTORE means, the wait to restore timer is active before transition to NORMAL. The status NOT-CONNECTED means, there is no timing reference signal connected to the timing port.

Type: Alphanumeric value set.

Range: NORMAL, FAILED, WAIT-TO-RESTORE, NOT-CONNECTED.

Default: NOT-CONNECTED.

g. **sclkref_lockstat** – station clock output timing reference lockout state

This parameter indicates the assigned station clock output timing reference lockout state. This parameter is only for SDH and for line timing references.

Type: Alphanumeric value set.

Range: LOCKOUT, NO-LOCKOUT.

Default: NO-LOCKOUT.

h. **syncref_pri** – system timing reference priority

This parameter defines the priority for assigned timing references.

When multiple input references have been given the same priority value, it implies non-revertive switching between the associated references. Otherwise, it implies revertive switching to the reference with the highest assigned priority. When the QL-control is disabled, the selection process is controlled by signal status and port priority. The greater the number, the lower the priority. Priority ″0″ means not assigned.

Type: Numeric value set.

Range: 0 (means DISABLE), 1, 2, 3, 4, 5, 6, 7, 8.

Default: 0 (means DISABLE).

i. **timing** – ASAP name

See definition in “RTRV-SYNCN: Retrieve synchronization” (p. 3-2955) section system timing of this parameter.

j. **`tpmode`** – timing port mode monitoring

This parameter specifies the monitoring status of the timing port. If the timing port is in the MONITORED mode and the reference is in FAILED status alarms are active for this port. The value NOT-MONITORED means that alarm monitoring is not occurring on the port.

Type: Alphanumeric value set.

Range: MONITORED, NOT-MONITORED.

Default: NOT-MONITORED.

k. **`tpql_in`** – incoming SSM value of the timing port

This parameter indicates the QL value for the timing port. The value range depends on the system synchronization characteristic.

The parameter describes the incoming SSM and takes the following values:

PRS – Stratum-1 Traceable STU – Synchronized Traceability Unknown

ST2 – Stratum-2

ST3 – Stratum-3

DUS/DNU – Don't Use for Synchronization

PRC – Primary Reference Clock

SSUT – Synchronization Supply Unit Transit node

SSUL – Synchronization Supply Unit Local node

SEC – SDH Equipment Clock

SSMn – other SSM values with invalid state (for SDH)

UNSTABLE – value is unstable

NONE – SSM value not supported by hardware

Type: Alphanumeric value set.

Range:

SONET: PRS (0001), STU (0000), ST2 (0111), ST3 (1010), DUS (1111), NONE

SDH: PRC (0010), SSUT (0100), SSUL (1000), SEC (1011), DNU (1111), SSM0 (0000), SSM1 (0001), SSM3 (0011), SSM5 (0101), SSM6 (0110), SSM7 (0111), SSM9 (1001), SSM10 (1010), SSM12 (1100), SSM13 (1101), SSM14 (1110), UNSTABLE, NONE.

Default: Not applicable.

l. **`tref_aid`** – timing reference assigned port AID

This parameter specifies the port AID of the assigned timing reference AID.

The parameter is rejected, if the timing reference already assigned to a timing port, or if the reference does not support SSM and timing port is provisioned AUTO, or if the unit which delivers the reference is not present or pre-provisioned. It is not possible to assign more then one reference from a unit. It is not possible to un-assign units from which a reference is assigned to a timing port. It is also not possible to assign a reference from a protection port in a protection group. The value NOT-CONNECTED means no reference is assigned to current timing port.

Type: AID.

Range: External_Timing_input_port_AID, SPortAID, NOT-CONNECTED.

Default: NOT-CONNECTED.

4. External Timing Input

   a. **`ext_if`** – external input signal format

      This parameter specifies the format of the external timing input.

      The value range depends on the system synchronization characteristic. The external timing input format can be set at both external timing inputs individually. The reference is rejected, if the reference does not support SSM and the timing port is provisioned AUTO.

      Type: Alphanumeric value set.

      Range:

      SONET: SF, ESF

      SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

      Default:

      SONET: ESF

      SDH: 2MHz.

   b. **`ext_isa`** – external input sa bit location

      This parameter defines the external input Sa bit location. It is only valid for the external timing input format “2Mbit/s” framed; i.e. for ports with format unequal to “2Mbit/s framed” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

      The location of the Sa bit is can be set at both external timing inputs individually. One of the Sa4 to Sa8 bits of the 2Mbit/s sub-multiframe is allocated for Synchronization Status Messages (SSM).

      Type: Alphanumeric value set.

      Range: SA4, SA5, SA6, SA7, SA8.

      Default: SA4.

   c. **`ext_issm`** – external input ssm support

      This parameter specifies if the SSM mode is supported or not on the external timing input port.

      The parameter is adjustable individually to both external timing input ports.

      Type: Alphanumeric value set.

      Range: SUPPORTED, NOT-SUPPORTED.

      Default: SONET: SUPPORTED, SDH: NOT-SUPPORTED.

5. External Timing Output

   a. **`ext_equ`** – cable equalization

      This parameter specifies the timing output cable equalization to adjust the specified timing reference output signal power. This parameter is valid only for DS1 (SONET).

      This parameter is rejected for all input interfaces except DS1. The values are represented by percentage (3.0dB=100%; 2.4dB=80%; 1.8dB=60%; 1.2dB=40%; 0.6dB=20%).

      Type: Numeric value set.

      Range: 20, 40, 60, 80, 100.

Default: 20.

b. **ext_of** – external output signal format

This parameter specifies the external output signal format. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: SF, ESF

SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

Default:

SONET: ESF

SDH: 2MHz.

c. **ext_osa** – external output sa bit location

This parameter defines the external output Sa bit location. This parameter is only valid for external timing output format “2Mbit/s framed”; i.e. for ports with format unequal to “2Mbit/s framed” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.

Range: SA4, SA5, SA6, SA7, SA8.

Default: SA4.

d. **ext_out_sig_stat** – external output signal status

This parameter indicates the external output signal status.

The value NORMAL indicates when the output timing signal is enabled and providing a correctly synchronized output timing signal. The value DISABLED indicates when the output timing signal has been disabled by means of user provisioning. The value UNACCEPTABLE indicates when the output timing signal is enabled but the timing output does not provide a suitable timing reference signal. This is caused by failed reference into the timing output process or references rejected based on threshold AIS criteria.

Type: Alphanumeric value set.

Range: NORMAL, DISABLED, UNACCEPTABLE.

Default: DISABLED.

e. **extout_en** – external timing output enabling/disabling

This parameter specifies whether the external timing output is enabled or disabled.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: DISABLE.

f. **extout_prov** – external timing output selection

This parameter specifies the operation mode of the external output timing.

The value SETG means the system clock. The value TLS means the output. For SONET both values are fixed to the respective external timing output. The value range depends on the system synchronization characteristic.

Type: Alphanumeric value set.

Range:
SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]
SDH: SETG, TLS.
Default: SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]
SDH: SETG.

g. **extout_ssm** – external timing output AIS mode

This parameter specifies which value will be send when the signal level is below the quality acceptance level or when *frcdus* is enabled. The parameter is valid depending from the system synchronization characteristic, for SDH 2Mbit/s framed and for SONET DS1 ESF; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

The value AISMODE means AIS is inserted in the external timing output signal. The value QLMODE means DUS is inserted in the external timing output signal.

Type: Alphanumeric value set.
Range: AISMODE, QLMODE.
Default: QLMODE.

h. **frcdus** – forced DNU/DUS

This parameter specifies the synchronization messaging forced DNU/DUS enable or disable for the timing output port. DUS applies to SONET and DNU applies to SDH. This is only valid for ’2Mbit/s framed’ and ’DS1 ESF’; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.
Range: ENABLE, DISABLE.
Default: DISABLE.

i. **loopdelay** – timing regenerator loop delay

This parameter indicates the delay in the SASE/BITS loop operation. The values are represented by seconds in steps of 1 seconds per loop.

Type: Numeric value set.
Range: 0…60 (1 seconds steps).
Default: 10.

j. **ql_out** – quality of the external timing output port

This parameter indicates the quality value on the signal for the timing output port.

Type: Alphanumeric value set.
Range:
SONET: PRS, STU, ST2, ST3, DUS
SDH: PRC, SSUL, SSUT, SEC, DNU.
Default: Not applicable.

k. **qlout_tais** – acceptance quality level for output threshold AIS

This parameter is the provisioned quality acceptance level for external timing output. When the incoming message is below this level, AIS or DUS (depending from the *extout_ssm* value) will be inserted into the external timing output. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: PRS, STU, ST2, ST3

SDH: PRC, SSUT, SSUL, SEC.

Default:

SONET: ST3

SDH: SEC.

l. **regen_loop** – timing regenerator loop

This parameter specifies the avoiding of loops over the SASE/BITS clocks.

When this parameter is set to ENABLED, the synchronization message DUS will be inserted in the traffic output port associated with the traffic input port used as source for the station clock. The parameter is selectable for both external timing outputs individually.

Type: Alphanumeric value set.

Range:

SONET: DISABLE, ENABLE

SDH: DISABLE, EXTREF1, EXTREF2, BOTH.

Default:

SONET: ENABLE

SDH: DISABLE.

m. **timing** – ASAP name

See definition in "RTRV-SYNCN: Retrieve synchronization" (p. 3-2955) section system timing of this parameter.

Detailed behavior description

The **RTRV-SYNCN** command retrieves the output parameters values for the specified input AID as described in "RTRV-SYNCN: Retrieve synchronization" (p. 3-2955) output parameters section.

Example input/output

The following is an example of the **RTRV-SYNCN** command:

```
RTRV-SYNCN:LT-UNITE:extrefall:123456;

   LT-UNITE 00-11-13 12:00:00
M  123456 COMPLD
   "EXTREF1:tref_aid=EXTTMG0,syncref_pri=2,ref_stat=NORMAL,
ref_lock=NO-LOCKOUT,inql=SSUT,inql_stat=NOT-SUPPORTED"
   "EXTREF2:tref_aid=EXTTMG1,syncref_pri=3,ref_stat=NORMAL,
ref_lock=NO-LOCKOUT,inql=SSUT,inql_stat=NOT-SUPPORTED"
```

;

Related TL1 messages

- “SET-SYNCN: Set synchronization” (p. 3-3158)
- “OPR-SYNCNSW: Operate synchronization switch ” (p. 3-2077)
- “RLS-SYNCNSW: Release synchronization switch” (p. 3-2215)

RTRV-T3: Retrieve DS3 port

The command **RTRV-T3** is used to retrieve all parameters related to the electrical DS3 port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-T3:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   Access Identifier of a DS3 port.
   Type: AID
   Range: APortAID of DS3 port. The keyword **all** is allowed up to shelf level.
3. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

In response to a valid **RTRV-T3** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the Output parameter section for the **RTRV-HDR** command.

Output parameters

Command parameters

1. **aid** – access identifier
   Type: AID
   Range: APortAID of DS3 port

Spec block parameters

1. **cbitmis** – C-bit Signal Format Mismatch
   DS3 C-bit Signal Format Mismatch parameter indicates a mismatch between the C-bit parity formatand the format of the incoming DS3 signal (ingress direction).

   The DS3 C-bit Signal Format Mismatch parameter shall have the value:

   - NORMAL, if the Signal Format (fmt) is set to C-BIT and a DS3 signal in C-bit parity channelized DS3 format is received
   - MISMATCH, if the Signal Format (fmt) is set to C-BIT and a DS3 signal in M23 channelized DS3 format is received
   - NA, if the Signal Format (fmt) is set to C-BIT and one of the alarms DS3LOS, DS3LOF, DS3AIS, DS3IDLE is present, or if the Signal Format (fmt) is set to another value than C-BIT, or if the value cannot be determined because pmode is set to AUTO or NMON or the circuit pack is not present or other system internal reasons.

   Type: Alphanumeric value set.
   Range: NORMAL, MISMATCH, NA.

2. **ds3pn** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.

3. **eif** – Electrical Interface Type
   The parameter eif indicates the type of input the port has been provisioned to receive.
   Type: Alphanumeric value set.
   Range: DS3

4. **feac** – Far-End Alarm and Control
   The parameter *feac* indicates the status of the FEAC channel of the incoming C-bit parity formatted DS3 signal. The parameter is only applicable, if *fmt* is set to C-BIT. The parameter shall have the value "NA" if the parameter is not applicable (*fmt* = M23 or UNFRAMED-CC), or the value for the parameter is not available because it cannot be determined.
   Type: Alphanumeric string.
   Range: String with 6 characters.

5. **`fmt`** – Signal Format
This parameter shows the format for the DS3 signal
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.
Range:
   - C-BIT (default): C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format
   - UNFRAMED-CC: UNFRAMED and CLEAR-CHANNEL format
6. **`fsflpbkstat`** – Far Side Facility Loopback Status
This parameter specifies the current far-side facility loopback status of the considered output aid. A far-side loopback means that the signal on the outgoing port is looped to the incoming port – this would thus correspond to loop via an external fibre cable, connecting the out-port with the in-port. If YES then the port is (far-side) loop-backed – if NO then there is no (far-side) loopback exististing for the aid.
Note: not both loopbacks (lpbk and fsflpbkstat) can be set to YES at the same time
Type: Alphanumeric value set.
Range: YES, NO (default)
7. **`infepm`** – Incoming DS3 Far End Path PM Enable
This parameter enables/disables the incoming DS3 far end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
8. **`inlnpm`** – Incoming DS3 Line PM Enable
This parameter enables/disables the incoming DS3 line performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
9. **`inpm`** – Incoming DS3 Near End Path PM Enable
This parameter enables/disables the incoming DS3 near end path performance monitoring.
Type: Alphanumeric value set.
Range: ENABLE, DISABLE (default).
10. **`lbo`** – Line Build Out
This parameter defines the cable length applicable for the DS3 port.
Type: Alphanumeric value set.
Range:
    - SHORT (default): 0 … 68.5 m (0 … 225 ft)
    - LONG: 68.5 … 137 m (225 … 450 ft).

11. **`lpbk`** – Loopback
    This parameter specifies the current near-side loopback status of the considered output *aid*.
    A near-side loopback means that the signal on the incoming port is looped to the outgoing port. If YES then the port is (near-side) loop-backed – if NO then there is no (near-side) loopback existing for the port.
    Type: Alphanumeric value set.
    Range:
    - YES
    - NO (default).
12. **`outmntc`** – Outgoing Maintenance Signal
    This parameter determines whether an AIS, a DS3AIS or IDLE signal is transmitted in response to an unconnected input.
    Type: Alphanumeric value set.
    Range: AIS (default), DS3AIS, IDLE
13. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
    This parameter enables/disables the outgoing DS3 far end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
14. **`outpm`** – Outgoing DS3 Near End Path PM Enable
    This parameter enables/disables the outgoing DS3 near end path performance monitoring.
    Type: Alphanumeric value set.
    Range: ENABLE, DISABLE (default).
15. **`pmode`** – Port Mode
    This parameter sets the mode of the port. In the case of AUTO the *pmode* state transitions to MON if the loss-of-signal alarm of this port is cleared.
    Type: Alphanumeric value set.
    Range: AUTO (default): indicates that, when a signal is detected at the port, then alarm monitoring will start. MON: indicates that alarm monitoring is taking place on the port. NMON: means that alarm monitoring is not occurring on the port.
16. **`pmsesl`** – PM Threshold for SES – Line
    PM Threshold (errors in a second) for declaring SES – Line
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44
17. **`pmsesp`** – PM Threshold for SES – Path Incoming
    PM Threshold (errors in a second) for declaring SES – Path in incoming direction.
    Type: numeric value set.
    Range: 1 … 8000
    Default: 44

18. **pmsespfe** – PM Threshold for Far-End SES – Path Incoming
PM Threshold (errors in a second) for declaring Far-End SES – Path in incoming direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44
19. **pmsespfeout** – PM Threshold (errors in a second) for declaring Far-End SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44
20. **pmsespout** – PM Threshold for SES – Path Outgoing
PM Threshold (errors in a second) for declaring SES – Path in outgoing direction.
Type: numeric value set.
Range: 1 … 8000
Default: 44
21. **ptype** – Port Type
This üparameter identifies the port type of the AID.
Type: Alphanumeric value set.
Range: T3
22. **sdthr** – Signal Degrade Threshold
Signal degrade threshold in a range from 10exp-3 to 10exp-9.
Type: numeric value set.
Range: (as exponents of 10): -3, -4, -5 -6 (default), -7, -8, -9
23. **servcond** – Service Condition
This parameter indicates its status concerning service involvement.
Type: Alphanumeric value set.
Range:
   - INSERVICE (default): Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.
24. **tca** – DS3 PM TCA
The parameter tca assigns an DS3 PM TCA profile name to the designated AID.
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the name additionally.
Range: String with a maximum of 24 characters, DEFAULT, DEFAULT0 (default).
25. **vmr** – Violation Monitoring Mode
This parameter indicates the status of the DS3 P-bit parity violation monitoring and removal.
Type: Alphanumeric value set. A hyphen (“-”) is allowed in the alphanumeric values additionally.
Range: VMR, VM, NO-VM (default)

Detailed behavior description

The command shall retrieve the parameters as described above. The **RTRV-T3** command do not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful execution of a **RTRV-T3** command. Not all output parameters are listed (missing ones indicated by …).

```
RTRV-T3:LT-UNITE:1-1-#-#-21-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   "1-1-#-#-21-1:cbitmis=NORMAL,eif=DS3,ptype=T3,fmt=C-BIT,.... sigmis=NORMAL,vmr=NO
      -VM"
;
```

Error responses

See definition in "RTRV-HDR: Retrieve header" (p. 3-2417) command error responses section for general validation rules. The error responses listed there also apply to the **RTRV-T3** command.

Validation rule: Check CCB against *ptype* and *eif*

Check wether the CCB fits to *ptype* and *eif*.

Error code: `ICNV – Input, Command Not Valid, Interface Type does not match`

Related TL1 messages

- "ED-T3: Edit DS3 port" (p. 3-1202)
- "ED-T3 notification" (p. 3-1210)
- "ENT-T3 notification" (p. 3-1890)
- "DLT-T3 notification" (p. 3-258)

References

GR-199-CORE (RTRV-*rr*)

RTRV-TACC: Retrieve test access

General

The **RTRV-TACC** command is used to retrieve the test access-related information on one or all of the test tributaries in the NE. Only the primary test tributary is retrieved.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): T1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-TACC**-*modifier:tid:aid:ctag;*

Input parameters

Command parameters

1. **modifier**
   This parameter indicates the cross-connection rate on which the retrieve test access command is to act.
   A distinction of the modifier usage has to be made between the retrieval (RTRV-TACC-modifier) and the other test access related commands (CONN-TACC-modifier, DISC-TACC-modifier, CHG-ACCMD-modifier):
   - The establishment (CONN-TACC-modifier), removal (DISC-TACC-modifier) or modification (CHG-ACCMD-modifier) of test access sessions is independent of the provisioned port interface-standard. I.e., either the SDH-rates (VC3, VC4 etc.) or the corresponding SONET-rates (STS1, STS3 etc.) can be used as modifier and no command validation between port interface-standard and modifier will be performed.
   - For the RTRV-TACC-modifier command the following holds: Again, no command validation between port interface-standard and modifier will be performed. The output parameter *rate*, however, is filled with the *rate* which was used with the CONN-TACC-modifier command.

   Type: Alphanumeric value set.
   Range:
   - ALL
   - SDH modifier: VC3, VC4, VC44C, VC416C, VC464C
   - SONET modifier: LOVT1, STS1, STS3, STS12, STS48, STS192.
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).

3. **`aid`** – Access Identifier
   AID of a SONET or SDH tributary. The keyword **`all`** is allowed on slot, port and tributary level.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`** values at all defined positions within the AID is allowed.
4. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

Output format

In response to a valid RTRV-TACC command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "tatrib1:rate[:spec_block]"
       ...
       ...
   "tatrib1:rate[:spec_block]"
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

The NE returns the following normal completion response for a successful RTRV-TACC command if no test access activity is found on the specified tributaries in the system:

```
   sid date time
M  ctag COMPLD
;
```

This output format is only valid if 'all' is used in the *aid*. Otherwise the validation rule leads to an error message.

Output parameters

Command parameters

1. **`tatrib1`** – primary Test Access Tributary
   AID of the primary test access tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`** values at any position of the aid is not allowed.
2. **`rate`** – Test Access rate
   This parameter specifies the rate of the signal that is being tested.
   Type: Alphanumeric value set.
   Range: VC3, VC4, VC44C, VC416C, VC464C, LOVT1, STS1, STS3, STS12, STS48, STS192.

Spec block parameters

1. **`eomode`** – output mode of the E-tributary
   This parameter is only present if the test session is created on a two-way cross-connection. The parameter is omitted for test sessions on one-way cross-connections or on idle E-tributaries.
   Type: Alphanumeric value set.
   Range:
   - NORM: this is the normal output i.e. the signal from the F-tributary is passed through to the E-tributary
   - AIS: this mode outputs an AIS signal.

   Default:
   - NORM, if the referring Ttest mode does not disconnect the output of the original cross-connection to this tributary
   - AIS, in all other case.
2. **`etrib`** – AID of the E-tributary
   AID of the E-tributary for the test session.
   Type: AID
   Range: GenTribAID or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.
3. **`fomode`** – output mode of the F-tributary
   This parameter is only present if the test session is created on a one-way or two-way cross-connection. The parameter is omitted for test sessions on idle E-tributaries.
   Type: Alphanumeric value set.
   Range:
   - NORM: this is the normal output i.e. the signal from the E-tributary is passed through to the F-tributary
   - AIS: this mode outputs an AIS signal.

Default:

- NORM, if the referring test mode does not disconnect the output of the original cross-connection to this tributary
- AIS, in all other cases.

4. **`ftrib`** – AID of the F-tributary
   AID of the F-tributary for the test session. If there is no cross-connection at the E-tributary, this parameter is omitted.
   Type: AID.
   Range: GenTribAID or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`**-values at any position of the aid is not allowed.
5. **`tatrib2`** – secondary Test Access Tributary
   AID of the secondary Test Access Tributary. In the case of test access modes with single test tributary the parameter is omitted.
   Type: AID.
   Range: STS1NumAID, PTNumAID, or GenLOTribAID.
   The usage of **`all`** or **`usall`** or **`vall`** values at any position of the aid is not allowed.
6. **`tstmde`** – Test Access Mode
   Type: Alphanumeric value set.

   Range: Allowed values:

   - MONE – Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.
   - MONF- Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.
   - MONEF – Monitor E-end and F-end mode of test access. This test access mode monitors (E end and F end) both directions of a two-way cross-connection.
   - SPLTE – Split access E-end mode of test access. This test access mode splits the connection between the E end and F end of a one-way cross-connection or a leg of two-way cross-connection. Can also be used in an idle tributary.
   - SPLTF – Split access F-end mode of test access. This test access mode splits the E-end and F-end connection of a two-way cross-connection to allow testing at the F-end.
   - SPLTEF – Split E-end and F-end mode of test access. This test access mode splits both directions of a two-way cross-connection to allow testing at both the E end and F end.

Example input/output

```
RTRV-TACC-STS1:LT-UNITE:1-1-#-#-3-1-1:123456;

   LT-UNITE 02-10-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-3-1-1:STS1:TSTMDE=MONEF,TATRIB2=1-1-#-#-4-1-1,ETRIB=1-1-#-#-6-1-
      1,EOMODE=NORM,FTRIB=1-1-#-#-7-1-1,FOMODE=NORM″
```

```
;
```

For Lower-order test access sessions:

```
RTRV-TACC-LOVT1:LT-UNITE:1-1-#-#-3-1-1-28:123456;

   LT-UNITE 02-10-01 08:00:00
M  123456 COMPLD
   "1-1-#-#-3-1-1-28:LOVT1:TSTMDE=MONEF,\
    TATRIB2=1-1-#-#-4-1-1-28,ETRIB=1-1-#-#-6-1-1-28,\
    EOMODE=NORM,FTRIB=1-1-#-#-7-1-1-28,FOMODE=NORM"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-TACC** command.

Validation rule: Tributary is not assigned as a test tributary

If the aid is not assigned as a test tributary and 'all' is not used in the aid, the following error message is returned:

Error code:

`SNVS – Status, Not in Valid State, Tributary not assigned as a test tributary`

Related TL1 messages

- "CONN-TACC: Connect test access" (p. 3-94)
- "CHG-ACCMD: Change access mode" (p. 3-65)
- "DISC-TACC: Disconnect test access" (p. 3-129)
- "RTRV-CRS: Retrieve cross-connection" (p. 3-2306)

RTRV-TCA-ASGNMT: Retrieve TCA assignment

The **RTRV-TCA-ASGNMT** command allows the user to retrieve a list of AIDs that have been assigned to the specified profile type and name.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-TCA-ASGNMT:*tid::ctag::pftype,pfname;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
4. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Output format

If the NE fully complies with the RTRV-TCA-ASGNMT request, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "pftype,pfname:aid"
   .     .     .
   .     .     .
;
```

Output parameters

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

1. **aid** – Access Identifier
   Type: AID
   Range: SPortAID, STSNumAID, EPortAID, VCGAID, VCGNumAID, PTNumAID, GenLOTribAID, TMuxAPortAID, TMuxAPTNumAID, DS1NumAID.
2. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
3. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Example input/output

The following example shows the successful completion of a RTRV-TCA-ASGNMT command by the NE:

```
  RTRV-TCA-ASGNMT:UNITE-NE::123456::section-line,ATT;
   UNITE-NE 00-01-01 08:00:00
```

```
M  123456 COMPLD
   ”section-line,ATT:1-1-#-#-22-1”
   ”section-line,ATT:1-1-#-#-23-1”
   ”section-line,ATT:1-1-#-#-24-1”
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-TCA-ASGNMT** command. See also subsection ’Partial Completion on RTRV Commands’ w.r.t. handling of internal error (e.g. communication to XC320 fails) for a subset of replies.

Validation rule: profile name does not exist

If the profile name specified does not exist.

Error code: `IDNV – Input, Data Not Valid, TCA profile does not exist`

Related TL1 messages

- “RTRV-TCA-PROF: Retrieve TCA profile” (p. 3-2982)
- “ENT-TCA-PROF: Enter TCA Profile” (p. 3-1894)
- “DLT-TCA-PROF: Delete TCA profile” (p. 3-259)
- “ED-TCA-PROF: Edit TCA profile” (p. 3-1214)

References

GR-833-CORE

RTRV-TCA-PROF: Retrieve TCA profile

General

The **RTRV-TCA-PROF** command allows the user to retrieve either a list of profile names that have been created for a specific profile type or the contents of a profile whose type and name are given in this command. The system supports two default sets of profiles called “DEFAULT” and “DEFAULT0”.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-TCA-PROF:*tid::ctag::pftype;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
4. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Output format

If the system receives this command and the profile name is NULL, the following response is returned:

```
   sid date time
M  ctag COMPLD
   "pftype:pfname"
   "pftype:pfname"
   "pftype:pfname"
   .     .     .     .     .     .
   .     .     .     .     .     .
;
```

If the system receives this command with a valid profile name, the following response is returned:

```
   sid date time
M  ctag COMPLD
   "pftype,pfname:spec_block"
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters - general profile

1. **pftype** – Profile Type
   This indicates the type of PM TCA profile for the selected facility.
   Type: value set.
   Range: SECTION-LINE, RS-MS, HOVC, PATH, ENET, ODU, LOVC, LOPATH, DS3, DS1.
2. **pfname** – Profile Name
   This indicates the name of the PM TCA profile which was already established by the command **ENT-TCA-PROF**. If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested. If a valid profile name is inserted in this field, the output will display the parameters within the profile with their values. A *pfname* value will always uniquely identify an existing profile for a specified *pftype* value. NULL means, that the Profile Name is omitted.
   Type: Alphanumeric string.
   Range: This is an alphanumeric character string with a maximum of 24 characters.
   Default: If this field is NULL, the output will only consist of a list of all profile names for the specific profile type that was requested.

Command parameters - profile parameters of type SECTION-LINE

1. **qhcvsoc3**
   15-minute threshold for CV-S bin for OC-3.
   Type: Integer.
   Range: 0-57599100
   Default: 140.
2. **daycvsoc3**
   Day threshold for CV-S bin of OC-3.
   Type: Integer.
   Range: 0-4294967295
   Default: 1344.
3. **qhcvsoc12**
   15-minute threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-57599100
   Default: 560.
4. **daycvsoc12**
   Day threshold for CV-S bin for OC-12.
   Type: Integer.
   Range: 0-4294967295
   Default: 5376.
5. **qhcvsoc48**
   15-minute threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-57599100
   Default: 2240.
6. **daycvsoc48**
   Day threshold for CV-S bin for OC-48.
   Type: Integer.
   Range: 0-4294967295
   Default: 21504.
7. **qhcvsoc192**
   15-minute threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-57599100
   Default: 8960.
8. **daycvsoc192**
   Day threshold for CV-S bin for OC-192.
   Type: Integer.
   Range: 0-4294967295
   Default: 86016.

9. **qhcvsoc768**
   15-minute threshold for CV-S bin for OC-768.
   Type: Integer.
   Range: 0-57599100
   Default: 35840.
10. **daycvsoc768**
    Day threshold for CV-S bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 344064.
11. **qhcvsec1**
    15-minute threshold for CV-S bin for EC-1.
    Type: Integer.
    Range: 0-57599100
    Default: 47
12. **daycvsec1**
    Day threshold for CV-S bin of EC-1
    Type: Integer.
    Range: 0-4294967295
    Default: 448
13. **qhess**
    15-minute threshold for ES-S bin.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **dayess**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **qhsess**
    15-minute threshold for SES-S bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
16. **daysess**
    Day threshold for ES-S bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.

17. **qhsefss**
    15-minute threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
18. **daysefss**
    Day threshold for SEFS-S bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.
19. **qhcvloc3**
    15-minute threshold for CV-L and CV-LFE bin for OC-3.
    Type: Integer.
    Range: 0-172799100
    Default: 140.
20. **daycvloc3**
    Day threshold for CV-L bin and CV-LFE for OC-3.
    Type: Integer.
    Range: 0-4294967295
    Default: 1344.
21. **qhcvloc12**
    15-minute threshold for CV-L and CV-LFE bin for OC-12.
    Type: Integer.
    Range: 0-691199100
    Default: 560.
22. **daycvloc12**
    Day threshold for CV-L bin and CV-LFE for OC-12.
    Type: Integer.
    Range: 0-4294967295
    Default: 5376.
23. **qhcvloc48**
    15-minute threshold for CV-L and CV-LFE bin for OC-48.
    Type: Integer.
    Range: 0-2764799100
    Default: 2240.
24. **daycvloc48**
    Day threshold for CV-L and CV-LFE bin for OC-48.
    Type: Integer.
    Range: 0-4294967295
    Default: 21504.

25. **`qhcvloc192`**
    15-minute threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 8960.
26. **`daycvloc192`**
    Day threshold for CV-L and CV-LFE bin for OC-192.
    Type: Integer.
    Range: 0-4294967295
    Default: 86016.
27. **`qhcvloc768`**
    15-minute threshold for CV-L and CV-LFE bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 35840.
28. **`daycvloc768`**
    Day threshold for CV-L and CV-LFE bin for OC-768.
    Type: Integer.
    Range: 0-4294967295
    Default: 344064.
29. **`qhcvlec1`**
    15-minute threshold for CV-L and CV-LFE bin for EC-1
    Type: Integer.
    Range: 0-57599100
    Default: 47
30. **`daycvlec1`**
    Day threshold for CV-L bin and CV-LFE for EC-1.
    Type: Integer.
    Range: 0-4294967295
    Default: 448
31. **`qhesl`**
    15-minute threshold for ES-L and ES-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 25.
32. **`dayesl`**
    Day threshold for ES-L and ES-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 250.

33. **`qhfeccloc192`**
    15-minute threshold for FECC-L bin for OC-192.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 5184.
34. **`dayfeccloc192`**
    Day threshold for FECC-L bin for OC-192.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 497644.
35. **`qhfeccloc768`**
    15-minute threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
36. **`dayfeccloc768`**
    Day threshold for FECC-L bin for OC-768.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.
37. **`qhsesl`**
    15-minute threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
38. **`daysesl`**
    Day threshold for SES-L and SES-LFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 40.
39. **`qhuasl`**
    15-minute threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
40. **`dayuasl`**
    Day threshold for UAS-L and UAS-LFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

Command parameters - profile parameters of type RS-MS

1. **qhrsnbbe**
   15-minute threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 9000.
2. **dayrsnbbe**
   Day threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
3. **qhrsnes**
   15-minute threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 180.
4. **dayrsnes**
   Day threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.
5. **qhrsnses**
   15-minute threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
6. **dayrsnses**
   Day threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 20.
7. **qhrsnuas**
   15-minute threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayrsnuas**
   Day threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 20.

9. **qhmsbbestm1**
   15-min threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 228000.
10. **daymsbbestm1**
    Day threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 432000.
11. **qhmsbbestm4**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 0-691199100
    Default: 912000.
12. **daymsbbestm4**
    Day threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
    Type: Integer.
    Range: 00-4294967295
    Default: 1728000.
13. **qhmsbbestm16**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-2764799100
    Default: 3648000.
14. **daymsbbestm16**
    Day threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 6912000.
15. **qhmsbbestm64**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 14592000.
16. **daymsbbestm64**
    Day threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 27648000.

17. **qhmsbbestm256**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 58368000.
18. **daymsbbestm256**
    Day threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 110592000.
19. **qhmses**
    15-minute threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 50.
20. **daymses**
    Day threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 150.
21. **qhmsnfeccstm64**
    15-minute threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 5184.
22. **daymsnfeccstm64**
    Day threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 49764.
23. **qhmsnfeccstm256**
    15-minute threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 20736.
24. **daymsnfeccstm256**
    Day threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 1990656.

25. **qhmsses**
    15-minute threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-810
    Default: 10.
26. **daymsses**
    Day threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-77760
    Default: 15.
27. **qhmsuas**
    15-min threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
28. **daymsuas**
    Day threshold for MS_N_UAS and MS_F_UAS for stm64 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Command parameters - reset profile parameters of type RS-MS

1. **qhrtrrsnbbe**
   15-minute reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **dayrtrrsnbbe**
   Day reset threshold for RS_N_BBE bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
3. **qhrtrrsnes**
   15-minute reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-900
   Default: 20.
4. **dayrtrrsnes**
   Day reset threshold for RS_N_ES bin.
   Type: Integer.
   Range: 0-86400
   Default: 150.

5. **qhrtrrsnses**
   15-minute reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **dayrtrrsnses**
   Day reset threshold for RS_N_SES bin.
   Type: Integer.
   Range: 0-77760
   Default: 0.
7. **qhrtrrsnuas**
   15-minute reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **dayrtrrsnuas**
   Day reset threshold for RS_N_UAS bin.
   Type: Integer.
   Range: 0-86400
   Default: 0.
9. **qhrtrmsbbestm1**
   15-min reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-172799100
   Default: 2400.
10. **dayrtrmsbbestm1**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm1 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 43200.
11. **qhrtrmsbbestm4**
   15-min reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 0-691199100
   Default: 9600.
12. **dayrtrmsbbestm4**
   Day reset threshold for MS_N_BBE and MS_F_BBE for stm4 bin.
   Type: Integer.
   Range: 0-4294967295
   Default: 172800.

13. **qhrtrmsbbestm16**
    15-min reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-2764799100
    Default: 38400.
14. **dayrtrmsbbestm16**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm16 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 691200.
15. **qhrtrmsbbestm64**
    15-min reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 153600.
16. **dayrtrmsbbestm64**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm64 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 2764800.
17. **qhrtrmsbbestm256**
    15-min threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 614400.
18. **dayrtrmsbbestm256**
    Day reset threshold for MS_N_BBE and MS_F_BBE for stm256 bin.
    Type: Integer.
    Range: 0-4294967295
    Default: 11059200.
19. **qhrtrmses**
    15-minute reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-900
    Default: 5.
20. **dayrtrmses**
    Day reset threshold for MS_N_ES and MS_F_ES bin.
    Type: Integer.
    Range: 0-86400
    Default: 15.

21. **`qhrtrmsnfeccstm64`**
    15-minute reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
22. **`dayrtrmsnfeccstm64`**
    Day reset threshold for MS-N-FECC bin forSTM-64.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
23. **`qhrtrmsnfeccstm256`**
    15-minute reset threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
24. **`dayrtrmsnfeccstm256`**
    Day reset threshold for MS-N-FECC bin forSTM-256.
    Type: Integer.
    Range: 0 – 4294967295
    Default: 0.
25. **`qhrtrmsses`**
    15-minute reset threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
26. **`dayrtrmsses`**
    Day reset threshold for MS_N_SES and MS_F_SES bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
27. **`qhrtrmsuas`**
    15-min reset threshold for MS_N_UAS and MS_F_UAS.
    Type: Integer.
    Range: 0-900
    Default: 0.
28. **`dayrtrmsuas`**
    Day reset threshold for MS_N_UAS and MS_F_UAS.
    Type: Integer.
    Range: 0-86400
    Default: 0.

Command parameters - profile parameters of type ODU

1. **qhodupbbeodu1**
   15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-18377100
   Default: 32000.
2. **dayodupbbeodu1**
   Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-1764201600
   Default: 61000.
3. **qhodupes**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 50.
4. **dayodupes**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 150.
5. **qhodupses**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 10.
6. **dayodupses**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 15.
7. **qhodupuas**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 15.
8. **dayodupuas**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 20.

Command parameters - reset profile parameters of type ODU

1. **`qhrtrodupbbeodu1`**
   15-minute threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-18377100
   Default: 200.
2. **`dayrtrodupbbeodu1`**
   Day threshold for ODU_N_BBE bin and ODU_F_BBE for ODU1P.
   Type: Integer.
   Range: 0-1764201600
   Default: 4800.
3. **`qhrtrodupes`**
   15-minute threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 5.
4. **`dayrtrodupes`**
   Day threshold for ODU_N_ES and ODU_F_ES bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 15.
5. **`qhrtrodupses`**
   15-minute threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-810
   Default: 0.
6. **`dayrtrodupses`**
   Day threshold for ODU_N_SES and ODU_F_SES bin for ODU1P.
   Type: Integer.
   Range: 0-77760
   Default: 0.
7. **`qhrtrodupuas`**
   15-minute threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-900
   Default: 0.
8. **`dayrtrodupuas`**
   Day threshold for ODU_N_UAS and ODU_F_UAS bin for ODU1P.
   Type: Integer.
   Range: 0-86400
   Default: 0.

Command parameters - profile parameters of type HO-VC

1. **qhvcbbevc3**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
2. **dayvcbbevc3**
   Day threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
3. **qhvcbbevc4**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbevc4**
   Day threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcbbevc44c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
6. **dayvcbbevc44c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
7. **qhvcbbevc416c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
8. **dayvcbbevc416c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.

9. **qhvcbbevc464c**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
10. **dayvcbbevc464c**
   Day threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
11. **qhvcesvc3**
   15-minute threshold for VC_N_ES and VC_F_ES VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 150.
12. **dayvcesvc3**
   Day threshold for VC_N_ES and VC_F_ES VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 600.
13. **qhvcesvc4**
   15-minute threshold for VC_N_ES and VC_F_ES for VC4 bin.
   Type: Integer.
   Range: 0-900
   Default: 180.
14. **dayvcesvc4**
   Day threshold for VC_N_ES and VC_F_ES for VC4 bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.
15. **qhvcesvc44c**
   15-minute threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
   Type: Integer.
   Range: 0-900
   Default: 180.
16. **dayvcesvc44c**
   Day threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
   Type: Integer.
   Range: 0-86400
   Default: 1500.

17. **qhvcesvc416c**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
18. **dayvcesvc416c**
    Day threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.
19. **qhvcesvc464c**
    15-minute threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
    Type: Integer.
    Range: 0-900
    Default: 180.
20. **dayvcesvc464c**
    Day threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
    Type: Integer.
    Range: 0-86400
    Default: 1500.
21. **qhvcsesvc3**
    15-minute threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
22. **dayvcsesvc3**
    Day threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
23. **qhvcsesvc4**
    15-minute threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
24. **dayvcsesvc4**
    Day threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.

25. qhvcsesvc44c
    15-minute threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
26. dayvcsesvc44c
    Day threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Range: 0-77760
    Default: 20.
27. **qhvcsesvc416c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
28. **dayvcsesvc416c**
    Day threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
29. **qhvcsesvc464c**
    15-minute threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
30. **dayvcsesvc464c**
    Day threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
31. **qhvcuasvc3**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
32. **dayvcuasvc3**
    Day threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

33. **qhvcuasvc4**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
34. **dayvcuasvc4**
    Day threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
35. **qhvcuasvc44c**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
36. **dayvcuasvc44c**
    Day threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
37. **qhvcuasvc416c**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
38. **dayvcuasvc416c**
    Day threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.
39. **qhvcuasvc464c**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
40. **dayvcuasvc464c**
    Day threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

41. **`qhpjcpos`**
    15-minute threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
42. **`daypjcpos`**
    Day threshold for positive PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
43. **`qhpjcneg`**
    15-minute threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
44. **`daypjcneg`**
    Day threshold for negative PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Command parameters - reset profile parameters of type HO-VC

1. **`qhrtrvcbbevc3`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
2. **`dayrtrvcbbevc3`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
3. **`qhrtrvcbbevc4`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **`dayrtrvcbbevc4`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC4 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.

5. **qhrtrvcbbevc44c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
6. **dayrtrvcbbevc44c**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC44C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
7. **qhrtrvcbbevc416c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
8. **dayrtrvcbbevc416c**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC416C bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
9. **qhrtrvcbbevc464c**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
10. **dayrtrvcbbevc464c**
    Day reset threshold for VC_N_BBE and VC_F_BBE for VC464C bin.
    Type: Integer.
    Range: 0-691113600
    Default: 4800.
11. **qhrtrvcesvc3**
    15-minute reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
12. **dayrtrvcesvc3**
    Day reset threshold for VC_N_ES and VC_F_ES VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 60.

13. **qhrtrvcesvc4**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
Type: Integer.
Range: 0-900
Default: 20.
14. **dayrtrvcesvc4**
Day reset threshold for VC_N_ES and VC_F_ES for VC4 bin.
Type: Integer.
Range: 0-86400
Default: 150.
15. **qhrtrvcesvc44c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-900
Default: 20.
16. **dayrtrvcesvc44c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-4C bin.
Type: Integer.
Range: 0-86400
Default: 150.
17. **qhrtrvcesvc416c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-900
Default: 20.
18. **dayrtrvcesvc416c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-16C bin.
Type: Integer.
Range: 0-86400
Default: 150.
19. **qhrtrvcesvc464c**
15-minute reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-900
Default: 20.
20. **dayrtrvcesvc464c**
Day reset threshold for VC_N_ES and VC_F_ES for VC4-64C bin.
Type: Integer.
Range: 0-86400
Default: 150.

21. **qhrtrvcsesvc3**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
22. **dayrtrvcsesvc3**
    Day reset threshold for VC_N_SES and VC_F_SES for VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
23. **qhrtrvcsesvc4**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
24. **dayrtrvcsesvc4**
    Day reset threshold for VC_N_SES and VC_F_SES for VC4 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
25. **qhrtrvcsesvc44c**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
26. **dayrtrvcsesvc44c**
    Day reset threshold for VC_N_SES and VC_F_SES for VC44C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
27. **qhrtrvcsesvc416c**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
28. **dayrtrvcsesvc416c**
    Day reset threshold for VC_N_SES and VC_F_SES for VC416C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.

29. **`qhrtrvcsesvc464c`**
    15-minute reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
30. **`dayrtrvcsesvc464c`**
    Day reset threshold for VC_N_SES and VC_F_SES for VC464C bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
31. **`qhrtrvcuasvc3`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
32. **`dayrtrvcuasvc3`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
33. **`qhrtrvcuasvc4`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
34. **`dayrtrvcuasvc4`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC4 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
35. **`qhrtrvcuasvc44c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
36. **`dayrtrvcuasvc44c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC44C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

37. **`qhrtrvcuasvc416c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
38. **`dayrtrvcuasvc416c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC416C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
39. **`qhrtrvcuasvc464c`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
40. **`dayrtrvcuasvc464c`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC464C bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
41. **`qhrtrpjcpos`**
    15-minute reset threshold for positive PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
42. **`dayrtrpjcpos`**
    Day reset threshold for positive PJC bin.
    Type: Integer.
    0-172800000
    Default: 0.
43. **`qhrtrpjcneg`**
    15-minute reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-1800000
    Default: 0.
44. **`dayrtrpjcneg`**
    Day reset threshold for negative PJC bin.
    Type: Integer.
    Range: 0-172800000
    Default: 0.

Command parameters - profile parameters of type PATH

1. **`qhcvpsts1`**
   15-minute threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-57599100
   Default: 15.
2. **`daycvpsts1`**
   Day threshold for CV-P and CV-PFE bin for STS1.
   Type: Integer.
   Range: 0-4294967295
   Default: 125.
3. **`qhcvpsts3c`**
   15-minute threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 0-57599100
   Default: 25.
4. **`daycvpsts3c`**
   Day threshold for CV-P and CV-PFE bin for STS3C.
   Type: Integer.
   Range: 4294967295
   Default: 250.
5. **`qhcvpsts12c`**
   15-minute threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 0-57599100
   Default: 75.
6. **`daycvpsts12c`**
   Day threshold for threshold for CV-P and CV-PFE bin for STS12C.
   Type: Integer.
   Range: 4294967295
   Default: 750.
7. **`qhcvpsts48c`**
   15-minute threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 0-57599100
   Default: 300.
8. **`daycvpsts48c`**
   Day threshold for CV-P and CV-PFE bin for STS48C.
   Type: Integer.
   Range: 4294967295
   Default: 3000.

9. **qhcvpsts192c**
   Day threshold for CV-P and CV-PFE bin for STS192C.
   Type: Integer.
   Range: 0-57599100
   Default: 1200.
10. **daycvpsts192c**
    Day threshold for CV-P and CV-PFE bin for STS192C.
    Type: Integer.
    Range: 0-4294967295
    Default: 12000.
11. **qhespsts1**
    15-minute threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-900
    Default: 12.
12. **dayespsts1**
    Day threshold for ES-P and ES-PFE bin for STS1.
    Type: Integer.
    Range: 0-86400
    Default: 100.
13. **qhespsts3c**
    15-minute threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-900
    Default: 20.
14. **dayespsts3c**
    Day threshold for ES-P and ES-PFE bin for STS3C.
    Type: Integer.
    Range: 0-86400
    Default: 200.
15. **qhespsts12c**
    15-minute threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-900
    Default: 60.
16. **dayespsts12c**
    Day threshold for ES-P and ES-PFE bin for STS12C.
    Type: Integer.
    Range: 0-86400
    Default: 600.

17. **`qhespsts48c`**
    15-minute threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-900
    Default: 240.
18. **`dayespsts48c`**
    Day threshold for ES-P and ES-PFE bin for STS48C.
    Type: Integer.
    Range: 0-86400
    Default: 2400.
19. **`qhespsts192c`**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-900
    Default: 240.
20. **`dayespsts192c`**
    Day threshold for ES-P and ES-PFE bin for STS192C.
    Type: Integer.
    Range: 0-86400
    Default: 9600.
21. **`qhsesp`**
    15-minute threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-810
    Default: 3.
22. **`daysesp`**
    Day threshold for SES-P and SES-PFE bin.
    Type: Integer.
    Range: 0-77760
    Default: 7.
23. **`qhuasp`**
    15-minute threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-900
    Default: 10.
24. **`dayuasp`**
    Day threshold for UAS-P and UAS-PFE bin.
    Type: Integer.
    Range: 0-86400
    Default: 10.

25. **`qhppjcgen`**
    15-minute threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
26. **`dayppjcgen`**
    Day threshold for PPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
27. **`qhnpjcgen`**
    15-minute threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
28. **`daynpjcgen`**
    Day threshold for NPJC-PGen bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
29. **`qhppjcdet`**
    15-minute threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
30. **`dayppjcdet`**
    Day threshold for PPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.
31. **`qhnpjcdet`**
    15-minute threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-1800000
    Default: 30.
32. **`daynpjcdet`**
    Day threshold for NPJC-PDet bin.
    Type: Integer.
    Range: 0-172800000
    Default: 2880.

Command parameters - profile parameters of type ENET

1. **`qhpdege`**
   15-minute threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-183000000
   Default: 187500.
2. **`daypdege`**
   Day threshold for PDE bin for GE.
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000.
3. **`qhpdevcg`**
   15-minute threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-183000000
   Default: 10.
4. **`daypdevcg`**
   Day threshold for PDE bin for VCG.
   Type: Integer.
   Range: 0-4294967295
   Default: 50.
5. **`qhedfe1ge`**
   15min EDFE for 1GbE Ethernet port
   Type: Integer.
   Range: 0-1371951220
   Default: 1875000
6. **`dayedfe1ge`**
   24 h EDFE for 1GbE Ethernet port
   Type: Integer.
   Range: 0-4294967295
   Default: 9000000
7. **`qhedfc1ge`**
   15min EDFC for 1GbE Ethernet port
   Type: Integer.
   Range: 0-1371951220
   Default: 20.
8. **`dayedfc1ge`**
   24 h EDFC for 1GbE Ethernet port
   Type: Integer.
   Range: 0-4294967295
   Default: 50.

9. **qhedfe10ge**
   15min EDFE for 10GbE Ethernet port
   Type: Integer.
   Range: 0-2147483647
   Default: 18750000.
10. **dayedfe10ge**
    24 h EDFE for 10GbE Ethernet port
    Type: Integer.
    Range: 0-4294967295
    Default: 90000000
11. **qhedfc10ge**
    15min EDFC for 10GbE Ethernet port
    Type: Integer.
    Range: 0-2147483647
    Default: 100.
12. **dayedfc10ge**
    24 h EDFC for 10GbE Ethernet port
    Type: Integer.
    Range: 0-4294967295
    Default: 500.
13. **qhedfe1gevcg**
    15min EDFE for 1GbE VCG port
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
14. **dayedfe1gevcg**
    24 h EDFE for 1GbE VCG port
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
15. **qhedfc1gevcg**
    15min EDFC for 1GbE VCG port
    Type: Integer.
    Range: 0-1371951220
    Default: 10.
16. **dayedfc1gevcg**
    24 h EDFC for 1GbE VCG port
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

17. **`qhedfe10gevcg`**
 15min EDFE for 10GbE VCG port
 Type: Integer.
 Range: 0-2147483647
 Default: 100.
18. **`dayedfe10gevcg`**
 24 h EDFE for 10GbE VCG port
 Type: Integer.
 Range: 0-4294967295
 Default: 500.
19. **`qhedfc10gevcg`**
 15min EDFC for 10GbE VCG port
 Type: Integer.
 Range: 0-2147483647
 Default: 100.
20. **`2.2.1.5.1.10.20`**
 24 h EDFC for 10GbE VCG port
 Type: Integer.
 Range: 0-4294967295
 Default: 500.

Command parameters - reset profile parameters of type ENET

1. **`qhrtrpdege`**
 15-minute reset threshold for PDE bin for GE.
 Type: Integer.
 Range: 0-183000000
 Default: 937500.
2. **`dayrtrpdege`**
 Day reset threshold for PDE bin for GE.
 Type: Integer.
 Range: 0-4294967295
 Default: 4500000.
3. **`qhrtrpdevcg`**
 15-minute reset threshold for PDE bin for VCG.
 Type: Integer.
 Range: 0-183000000
 Default: 2.
4. **`dayrtrpdevcg`**
 Day reset threshold for PDE bin for VCG.
 Type: Integer.
 Range: 0-4294967295
 Default: 5.

5. **`qhrtredfe1ge`**
   15min reset threshold for EDFE of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 937500.
6. **`dayrtredfe1ge`**
   24h reset threshold for EDFE of 1GbE Ethernet port
   Type: Integer.
   Range: 0-4294967295
   Default: 4500000.
7. **`qhrtredfc1ge`**
   15min reset threshold for EDFC of 1GbE Ethernet port.
   Type: Integer.
   Range: 0-1371951220
   Default: 2.
8. **`dayrtredfc1ge`**
   24h reset threshold for EDFC of 1GbE Ethernet port
   Type: Integer.
   Range: 0-4294967295
   Default: 5.
9. **`qhrtredfe10ge`**
   15min reset threshold for EDFE of 10GbE Ethernet port
   Type: Integer.
   0-2147483647
   Default: 9375000
10. **`dayrtredfe10ge`**
    24h reset threshold for EDFE of 10GbE Ethernet port
    Type: Integer.
    Range: 0-4294967295
    Default: 45000000.
11. **`qhrtredfc10ge`**
    15min reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    0-2147483647
    Default: 20.
12. **`dayrtredfc10ge`**
    24h reset threshold for EDFC of 10GbE Ethernet port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

13. **`qhrtredfe1gevcg`**
    15min reset threshold for EDFE of 1GbE VCG por
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
14. **`dayrtredfe1gevcg`**
    24h reset threshold for EDFE of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
15. **`qhrtredfc1gevcg`**
    15min reset threshold for EDFC of 1GbE VCG port
    Type: Integer.
    Range: 0-1371951220
    Default: 2.
16. **`dayrtredfc1gevcg`**
    24h reset threshold for EDFC of 1GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 5.
17. **`qhrtredfe10gevcg`**
    15min reset threshold for EDFE of 10GbE VCG port
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
18. **`dayrtredfe10gevcg`**
    24h reset threshold for EDFE of 10GbE VCG port.
    Type: Integer.
    Range: 0-4294967295
    Default: 50.
19. **`qhrtredfc10gevcg`**
    15min reset threshold for EDFC of 10GbE VCG port.
    Type: Integer.
    Range: 0-2147483647
    Default: 20.
20. **`dayrtredfc10gevcg`**
    24h reset threshold for EDFC of 10GbE VCG port
    Type: Integer.
    Range: 0-4294967295
    Default: 50.

Command parameters - profile parameters of type LOVC

1. **qhvcbbevc12**
   15-minute threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 9000.
2. **dayvcbbevc12**
   Day threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 12000.
3. **qhvcbbelovc3**
   15minute threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 36000.
4. **dayvcbbelovc3**
   Day threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 48000.
5. **qhvcesvc12**
   15-minute threshold for VVC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 120.
6. **dayvcesvc12**
   Day threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 350.
7. **qhvceslovc3**
   15-minute threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 150.
8. **dayvceslovc3**
   Day threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 600.

9. **qhvcsesvc12**
   15-minute threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 15.
10. **dayvcsesvc12**
    Day threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
11. **qhvcseslovc3**
    15-minute threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 15.
12. **dayvcseslovc3**
    Day threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 20.
13. **qhvcuasvc12**
    15-minute threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 15.
14. **dayvcuasvc12**
    Day threshold for VC_N_UAS and VC_F_UAS for VC12 bin..
    Type: Integer.
    Range: 0-86400
    Default: 20.
15. **qhvcuaslovc3**
    15-minute threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin..
    Type: Integer.
    Range: 0-900
    Default: 15.
16. **dayvcuaslovc3**
    Day threshold for VC_N_UAS and C_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 20.

Command parameters - reset profile parameters of type LOVC

1. **`qhrtrvcbbevc12`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-1799100
   Default: 50.
2. **`dayrtrvcbbevc12`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for VC12 bin.
   Type: Integer.
   Range: 0-172713600
   Default: 1200.
3. **`qhrtrvcbbelovc3`**
   15-minute reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-7199100
   Default: 200.
4. **`dayrtrvcbbelovc3`**
   Day reset threshold for VC_N_BBE and VC_F_BBE for lower order VC3 bin.
   Type: Integer.
   Range: 0-691113600
   Default: 4800.
5. **`qhrtrvcesvc12`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-900
   Default: 5.
6. **`dayrtrvcesvc12`**
   Day reset threshold for VC_N_ES and VC_F_ES for VC12 bin.
   Type: Integer.
   Range: 0-86400
   Default: 35.
7. **`qhrtrvceslovc3`**
   15-minute reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-900
   Default: 10.
8. **`dayrtrvceslovc3`**
   Day reset threshold for VC_N_ES and VC_F_ES for lower order VC3 bin.
   Type: Integer.
   Range: 0-86400
   Default: 60.

9. **`qhrtrvcsesvc12`**
   15-minute reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
   Type: Integer.
   Range: 0-810
   Default: 0.
10. **`dayrtrvcsesvc12`**
    Day reset threshold for VC_N_SES and VC_F_SES for VC12 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
11. **`qhrtrvcseslovc3`**
    15-minute reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-810
    Default: 0.
12. **`dayrtrvcseslovc3`**
    Day reset threshold for VC_N_SES and VC_F_SES for lower order VC3 bin.
    Type: Integer.
    Range: 0-77760
    Default: 0.
13. **`qhrtrvcuasvc12`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
14. **`dayrtrvcuasvc12`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for VC12 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.
15. **`qhrtrvcuaslovc3`**
    15-minute reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-900
    Default: 0.
16. **`dayrtrvcuaslovc3`**
    Day reset threshold for VC_N_UAS and VC_F_UAS for lower order VC3 bin.
    Type: Integer.
    Range: 0-86400
    Default: 0.

Command parameters - profile parameters of type LOPATH

1. **`qhcvvt1`**
   15-minute threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-14399100
   Default: 75.
2. **`daycvvt1`**
   Day threshold for CV-P and CV-PFE bin for VT1.
   Type: Integer.
   Range: 0-1382313600
   Default: 750.
3. **`qhesvt1`**
   15-minute threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
4. **`dayesvt1`**
   Day threshold for ES-P and ES-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.
5. **`qhsesvt1`**
   15-minute threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-810
   Default: 40.
6. **`daysesvt1`**
   Day threshold for SES-P and SES-PFE bin for VT1.
   Type: Integer.
   Range: 0-77760
   Default: 900.
7. **`qhuasvt1`**
   15-minute threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-900
   Default: 40.
8. **`dayuasvt1`**
   Day threshold for UAS-P and UAS-PFE bin for VT1.
   Type: Integer.
   Range: 0-86400
   Default: 900.

Command parameters - profile parameters of type DS3

1. **`qhcvl`**
   15-minute threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0-7200000
   Default: 387.
2. **`daycvl`**
   Day threshold for CV-L bin of DS3 Line.
   Type: Integer.
   Range: 0- 691200000
   Default: 3865.
3. **`qhesl`**
   15-minute threshold for Es-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
4. **`dayesl`**
   Day threshold for ES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
5. **`qhesal`**
   15-minute threshold for ESA-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
6. **`dayesal`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.
7. **`qhesbl`**
   15-minute threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-900
   Default: 25.
8. **`dayesbl`**
   Day threshold for ESB-L bin of DS3 Line.
   Type: Integer.
   Range: 0-86400
   Default: 250.

9. **qhsesl**
   15-minute threshold for SES-L bin of DS3 Line.
   Type: Integer.
   Range: 0-810
   Default: 4.
10. **daysesl**
    Day threshold for SES-L bin of DS3 Line.
    Type: Integer.
    Range: 0-77760
    Default: 40.
11. **qhcvp**
    15-minute threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-7200000
    Default: 382.
12. **daycvp**
    Day threshold for CV-P and CV-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-691200000
    Default: 3820.
13. **qhesp**
    15-minute threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.
14. **dayesp**
    Day threshold for ES-P and ES-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 250.
15. **qhesap**
    15-minute threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 25.

16. **dayesap**
Day threshold for ESA-P and ESA-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 250.
17. **qhesbp**
15-minute threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-900
Default: 25.
18. **dayesbp**
Day threshold for ESB-P and ESB-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 250.
19. **qhsesp**
15-minute threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-810
Default: 4.
20. **daysesp**
Day threshold for SES-P and SES-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-77760
Default: 40.
21. **qhuasp**
15-minute threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-900
Default: 10.
22. **dayuasp**
Day threshold for UAS-P and UAS-PFE bin of DS3 Path ingress and egress direction.
Type: Integer.
Range: 0-86400
Default: 10.

23. **qhsasp**
    15-minute threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
24. **daysasp**
    Day threshold for SAS-P and SAS-PFE bin of DS3 Path ingress and egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 8.

Command parameters - profile parameters of type DS1

1. **qhcvsfp**
   15-minute threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 72.
2. **daycvsfp**
   Day threshold for CV-P bin of DS1 Path with SF format in egress direction.
   Type: Integer.
   Range: 0- 1048575
   Default: 691.
3. **qhcvesfp**
   15-minute threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-16383
   Default: 13296.
4. **daycvesfp**
   Day threshold for CV-P bin of DS1 Path with ESF format in egress direction.
   Type: Integer.
   Range: 0-1048575
   Default: 132960.
5. **qhesp**
   15-minute threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 65.
6. **dayesp**
   Day threshold for ES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 648.

7. **qhsesp**
   15-minute threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-810
   Default: 10.
8. **daysesp**
   Day threshold for SES-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-86400
   Default: 250.
9. **qhuasp**
   15-minute threshold for UAS-P bin of DS1 Path egress direction.
   Type: Integer.
   Range: 0-900
   Default: 10.
10. **dayuasp**
    Day threshold for UAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 10.
11. **qhsasp**
    15-minute threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-900
    Default: 2.
12. **daysasp**
    Day threshold for SAS-P bin of DS1 Path egress direction.
    Type: Integer.
    Range: 0-86400
    Default: 17.

Detailed behavior description

The command shall retrieve the list of profile names of the specified profile type, if only the profile type is specified in the command. If profile type and profile name is specified, and a profile with the specified name exists, then the parameters of the profile is retrieved. If the profile does not exist, the system shall not return a response to the command.

Example input/output

The following example shows the successful completion of a RTRV-TCA-PROF command by the NE:

```
RTRV-TCA-PROF:UNITE-NE::123456::section-line;
```

```
   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
   "SECTION-LINE:DEFAULT"
   "SECTION-LINE:DEFAULT0"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-TCA-PROF** command.

Related TL1 messages

- “DLT-TCA-PROF: Delete TCA profile” (p. 3-259)
- “ED-TCA-PROF: Edit TCA profile” (p. 3-1214)
- “ENT-TCA-PROF: Enter TCA Profile” (p. 3-1894)

References

GR-833-CORE

RTRV-TMAT3: Retrieve asynchronous adjunct port of the TransMUX card

General

The command **RTRV-TMAT3** allows the user to retrieve all parameters related to the TransMUX asynchronous adjunct port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-TMAT3:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier of a TransMUX asynchronous adjunct port.
   Type: AID.
   Range: TMuxAPortAID.
   The keyword **all** is allowed up to shelf level.
   Default: None.
3. **ctag** – Correlation Tag
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

In response to a valid **RTRV-TMAT3** command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Access Identifier.
   Type: AID.
   Range: TMuxAPortAID. The keyword **`all`** is not allowed.
   Default: None.

Spec block parameters

1. **`cbitmis`** – C-bit Signal Format Mismatch
   DS3 C-bit Signal Format Mismatch parameter indicates a mismatch between the provisioned C-bit signal format and the C-bit format of the DS3 signal.
   For the TransMUX Card: – outgoing DS3 signal (egress direction).
   The DS3 C-bit Signal Format Mismatch parameter will show a MISMATCH value if the DS3 port were provisioned for C-bit parity format, but it actually receives a different type of DS3 format. Otherwise, it will show the value NORMAL.
   The DS3 C-bit Signal Format Mismatch parameter shall have the value:
   - NORMAL, if the Signal Format (fmt) is set to C-BIT and a DS3 signal in C-bit parity channelized DS3 format is received
   - MISMATCH, if the Signal Format (fmt) is set to C-BIT and a DS3 signal in M23 channelized DS3 format is received
   - NA, if the Signal Format (fmt) is set to C-BIT and one of the alarms DS3LOS, DS3LOF, DS3AIS, DS3IDLE is present, or if the Signal Format (fmt) is set to another value than C-BIT, or if the value cannot be determined because pmode is set to NMON or the circuit pack is not present or other system internal reasons.

   Type: Alphanumeric value set.
   Range: NORMAL; MISMATCH; NA.
   Default: None.
2. **`ds3pn`** – DS3 ASAP
   DS3 Alarm Severity Assignment Profile. This parameter retrieves the ASAP profile name of type DS3 to which the designated AID points.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: String with a maximum of 24 characters.
   Default: DEFAULT.

3. **`feac`** – Far-End Alarm and Control
   The parameter *feac* indicates the status of the FEAC channel of the incoming C-bit parity formatted DS3 signal. The parameter is only applicable, if *fmt* is set to C-BIT. The parameter shall have the value "NA" if the parameter is not applicable (*fmt* = M23 or UNFRAMED-CC), or the value for the parameter is not available because it cannot be determined.
   For the TransMUX card, the FEAC channel of the outgoing DS3 signal (egress direction) is considered.
   Type: Alphanumeric string.
   Range: String with 6 characters.
   Default: None.

4. **`fmt`** – Signal Format
   This parameter shows the format for the DS3 signal
   Type: Alphanumeric value set. A hyphen ("-") is allowed in the alphanumeric values additionally.

   Range:

   - C-BIT: C-bit parity channelized DS3 format
   - M23: M23 channelized DS3 format. M23 format contains seven DS2 signals, which contain DS1 signals.

   Default: C-BIT.

5. **`outfepm`** – Outgoing DS3 Far End Path PM Enable
   This parameter enables/disables the outgoing DS3 far end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.

6. **`outpm`** – Outgoing DS3 Near End Path PM Enable
   This parameter enables/disables the outgoing DS3 near end path performance monitoring.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.

7. **`pmode`** – Port Mode
   This parameter sets the mode of the port. In the case of AUTO the pmode state transitions to MON if the loss-of-signal alarm of this port is cleared.
   Type: Alphanumeric value set.

   Range:

   - MON indicates that alarm monitoring is taking place on the port.
   - NMON means that alarm monitoring is not occurring on the port.

   Default: NMON.

8. **ptype** – Port Type
   This identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range: TMAT3
   Default: TMAT3
9. **servcond** – Service Condition
   This parameter indicates its status concerning service involvement.
   Type: Alphanumeric value set.
   Range:
   - INSERVICE: Provisioning values for service are accepted
   - OUTOFSERVICE: Provisioning values for service are not accepted.

   Default: INSERVICE.
10. **tca** – DS3 PM TCA
    This parameter assigns a DS3 PM TCA profile name to the designated AID.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: String with a maximum of 24 characters.
    - DEFAULT
    - DEFAULT0

    Default: DEFAULT.

Detailed behavior description

The command retrieves the parameters as described above. The **RTRV-TMAT3** command does not generate a **REPT DBCHG** message.

Example input/output

The following example shows the successful completion of a **RTRV-TMAT3** command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-TMAT3:UNITE-NE:1-1-#-#-21-tma1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   ″1-1-#-#-21-tma1:cbitmis=NORMAL,ptype=TMAT3,fmt=C-BIT,....,tca=DEFAULT″
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **RTRV-TMAT3** command.

Related TL1 messages

- "ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card" (p. 3-1306)
- "ED-TMAT3 notification" (p. 3-1310)
- "ENT-TMAT3 notification" (p. 3-1941)
- "DLT-TMAT3 notification" (p. 3-263)

References

GR-199-CORE (RTRV-rr)

RTRV-TMST3: Retrieve synchronous adjunct port of the TransMUX card

General

The command **RTRV-TMAT3** allows the user to retrieve all parameters related to the TransMUX synchronous adjunct port.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax:

RTRV-TMST3:*tid:aid:ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier of a TransMUX virtual synchronous adjunct port.
   Type: AID
   Range: TMuxSPortAID
   The keyword **all** is allowed up to shelf level.
   Default: None
3. **ctag** – Correlation Tag
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

In response to a valid RTRV-TMST3 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   ″aid[:spec_block]″
       ...
       ...
   ″aid[:spec_block]″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   Access Identifier.
   Type: AID.
   Range: TMuxSPortAID. The keyword **all** is not allowed.
   Default: None.

Spec block parameters

1. **ptype** – Port Type
   This identifies the port type of the AID.
   Type: Alphanumeric value set.
   Range: TMST3
   Default: TMST3

Detailed behavior description

The command retrieves the parameters as described above. The command **RTRV-TMST3** does not generate a REPT DBCHG message.

Example input/output

The following example shows the successful completion of a RTRV-TMST3 command. Not all output parameters are listed (missing ones indicated by …):

```
RTRV-TMST3:LT-UNITE:1-1-#-#-21-tms1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   "1-1-#-#-21-tms1:ptype=TMST3"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-TMST3 command.

Related TL1 messages

- "ENT-TMST3 notification" (p. 3-1943)
- "DLT-TMST3 notification" (p. 3-264)

References

GR-199-CORE (RTRV-rr)

RTRV-TMUXPM: Retrieve performance monitoring data of TransMUX pack

The **RTRV-TMUXPM** command is initiated by a user to request the network element to send the current and/or historical Performance Monitoring (PM) data associated with performance monitoring points located on a TransMUX pack. For each value in the *montype* parameter of the **RTRV-TMUXPM** input parameters section, PM counter values will be stored in the current 15-minute period, the 32 previous 15-minute periods (past 8 hours worth of data), current day, and previous day periods.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): PM1

Priority: 1

Abortable: Yes.

Input format

Command syntax:
RTRV-TMUXPM-*modifier*:*tid*:*aid*:*ctag*::*[montype]*,*[monlev]*,*[locn]*,,*tmper*,*[mondat]*,*[montm]*;

Input parameters

Command parameters

1. **modifier**
This parameter defines the type of performance monitoring point.
Type: Alphanumeric value set.
Range: The following value is for SONET only: TMAT3, TMASTS1, DS1.
Default: None.
2. **tid** – Target Identifier
See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
3. **aid** – Access Identifier
Type: AID
Range:
For TMAT3 modifier: TMuxAPortAID
For TMASTS1 modifier: TMuxAPTNumAID
For DS1 modifier: DS1NumAID
Wildcard values in parts of both AIDs are usable as follows:
   - *all*, *usall* in Simple_UnivSlotAID,
   - *all* in Simple_TMuxAPortAID,

- *all* in Simple_TMuxAPTNumAID,
- *all* in Simple_DS1NumAID.

Default: None.

4. **`ctag`** – Correlation Tag
See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).

5. **`montype`** – Monitored parameter type
This is the monitored parameter type for which data is requested. If the value is omitted, this parameter defaults to ALL.
Type: value set.
Range:
   - The following values are for TMASTS1 modifier: CVP, ESP, SESP, UASP, FC-P, CV-PFE, ES-PFE, SES-PFE, UAS-PFE, FC-PFE
   - The following values are for TMAT3 modifier: CVP-EGR, ESP-EGR, ESA-P-EGR, ESB-P-EGR, SESP-EGR, UASP-EGR, FC-P-EGR, SEFS-P-EGR, SAS-P-EGR, AISS-P-EGR, CV-PFE-EGR, ES-PFE-EGR, ESA-PFE-EGR, ESB-PFE-EGR, SES-PFE-EGR, UAS-PFE-EGR, FC-PFE-EGR, SAS-PFE-EGR
   - The following values are for DS1 modifier: CVP-EGR, ESP-EGR, SESP-EGR, UASP-EGR, SAS-P-EGR, AISS-P-EGR

   The following value is also allowed for all input modifiers: ALL.
   Default: ALL.

6. **`monlev`** – Monitored Level
This parameter provides the level for the monitored PM parameter. The parameter is only evaluated if the value of *montype* parameter is ALL.
Type: value set.
Range:
   - 1-UP: Means the NE will report all non-zero data and all data which is marked as non-valid (parameter validity).
   - 0-UP: Means the NE will report zero and all positive data.

   Default: 1-UP.

7. **`locn`** – Location
Location for which the PM data is monitored. The parameter value is only considered if the *montype* value is ALL; otherwise it is ignored for the calculation of the command reply (i.e has no impact on the command).
Type: value set.
Range:
   - NEND: Near-End PM data is requested.
   - FEND: Far-End PM data is requested.
   - ALL: All near-end and far-end data is requested. It applies only to input direction.

   Default: ALL.

8. **tmper** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.

   Default: None.
9. **mondat** – Monitored Date
   This requests the beginning date of the interval for which the PM data is to be reported. For both values of tmper parameter, the maximum allowable range of valid dates includes the current and previous one day.
   Type: value set.
   Range:
   - MM-DD: Month-Day. A leading zero (first M or first D) is optional and may be omitted.
   - ALL: Current and previous day.

   Default: Current date
10. **montm** – Monitored Time
    This parameter specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. The user is able to specify a monitored time in three modes:
    a. A monitored time interval (i.e., from hour-minutes to hour-minutes: HOD-MOH&&THOD-TMOH).
    b. A single time interval (i.e., the hour and minutes for the appropriate 15 minute data: HOD-MOH).
    c. All applicable beginning times for the given *mondat* value (i.e., all).

    Where HOD-MOH represents the start of the requested PM retrieval period (hours and minutes) and THOD-TMOH represents the end of the requested PM retrieval period. The value ALL must be used if the *mondat* value provided is ALL.
    Type: Character string.

Range:

- HOD-MOH: This specifies the beginning time of day of the requested PM period that was set by *tmper* parameter. This parameter may be omitted or have one of the following values: Hour Of Day- Minute Of Hour, where HOD ranges from 0 to 23 and MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.
- THOD-TMOH: To Hour Of Day-To Minute Of Hour, where THOD ranges from 0 to 23 and TMOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0. Parameter grouping is used when specifying THOD-TMOH. For example, 08-15&&09-00 specifies the HOD-MOH and THOD-TMOH of 08-15 and 09-00, respectively. THOD-TMOH of $$-$$ indicates the current Hour Of Day-Minute Of Hour. THOD-TMOH of ##- ## indicates the previous 15-min interval.
- ALL: All applicable beginning times for the given *mondat* value.

If the *tmper* value is 1-DAY, the *montm* parameter will not be used.
Default:
If THOD-TMOH is not specified, only the PM data for the retrieval period specified by HOD-MOH will be retrieved.
If HOD-MOH is omitted, the PM period for the current time will be reported.

Output format

Normal completion message

If the network element fully complies with the **RTRV-TMUXPM** request, the following normal completion response is returned:

```
    sid date time
M   ctag COMPLD
   "aid:montype,monval,[validity],[tca],[locn],,tmper,[mondat],[montm]"
   . . . . .
   . . . . .
   . . . . .
;
```

Note: The trailing comma has to be present even if the optional parameter is not supported for a dedicated release.

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Normal completion message – empty reply

If there is no PM data stored within the NE, then the following response is returned:

```
   sid date time
M   ctag COMPLD
```

;

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Type: AID
   Range:
   TMuxAPortAID, TMuxAPTNumAID, or DS1NumAID
   Wildcard values, for example *all*, *usall*, are not used in any parts of the AIDs.
   Default: None.
2. **`montype`** – Monitored parameter type
   This is the monitored parameter type for which data is requested. If the value is omitted, this parameter defaults to ALL.
   Type: value set.
   Range:
   - The following values are for TMASTS1 modifier: CVP, ESP, SESP, UASP, FC-P, CV-PFE, ES-PFE, SES-PFE, UAS-PFE, FC-PFE
   - The following values are for TMAT3 modifier: CVP-EGR, ESP-EGR, ESA-P-EGR, ESB-P-EGR, SESP-EGR, UASP-EGR, FC-P-EGR, SEFS-P-EGR, SAS-P-EGR, AISS-P-EGR, CV-PFE-EGR, ES-PFE-EGR, ESA-PFE-EGR, ESB-PFE-EGR, SES-PFE-EGR, UAS-PFE-EGR, FC-PFE-EGR, SAS-PFE-EGR
   - The following values are for DS1 modifier: CVP-EGR, ESP-EGR, SESP-EGR, UASP-EGR, SAS-P-EGR, AISS-P-EGR

   The following value is also allowed for all input modifiers: ALL.
   Default: ALL.
3. **`monval`** – Monitored Value
   This parameter contains the measured value of the parameter specified in montype.
   Type: Integer.
   Range: 0-4294967295
   The specified range is the maximal range taking all PM parameters into consideration. Specific PM parameters may have a smaller range.
   W.r.t. value range for the specific PM parameters see
   - “Output parameters” section in “RTRV-TCA-PROF: Retrieve TCA profile” (p. 3-2982) (if thresholding is applicable for that parameter)
   - PM parameter introduction section (if thresholding is not applicable for that parameter).
4. **`validity`** – validity indicator
   This parameter indicates the validity of the PM data.
   Type: Integer.
   Range: 1: valid; (omitted): non-valid.
   If the value is omitted, it indicates that the validity is ″non valid″.

5. **`tca`** – Threshold Crossing Alert
   Type: value set.
   Range:
   For the current interval: *tca* indicates, that the current value exceeds the provisioned threshold. If the value is omitted, it indicates that the current value is below the provisioned threshold.
   For previous intervals, the *tca* parameter indicates the state at the end of the interval.
   Default: None.
6. **`locn`** – Location
   Location for which the PM data is monitored.
   Type: value set.
   Range:
   - NEND: Near-End PM data is requested
   - FEND: Far-End PM data is requested.
7. **`tmper`** – Time Period
   This requests PM data for a specified time interval.
   Type: value set.
   Range:
   - 15-MIN: This requests PM data in 15-minute intervals
   - 1-DAY: This requests daily PM data.

   Default: None.
8. **`mondat`** – Monitored Date
   This parameter specifies the beginning date of the interval for which the PM data is reported. The parameter can be omitted in case of current date.
   Type: value set.
   Range: MM-DD: Month-Day
   Default: Current date
9. **`montm`** – Monitored Time
   This parameter specifies the beginning time of day for the reported pm register. The parameter is optional and may be omitted for the reporting of current bins (registers).
   Type: Character string.
   Range: HOD-MOH: HOD ranges from 0 to 23; MOH is 0, 15, 30 and 45. A leading zero is optional. For example, 00 means 0.

Detailed behavior description

Fault tolerant and filtered retrievals

This command shall only return those parameters, which are specified by the input parameters. This command shall omit parts of the response (maybe to return an empty response) if related performance monitoring has not been enabled (independent whether this occurs due to a specific or wildcard query). This command shall oomitparts of the response due to a wildcard query (maybe to return an empty response) if the related entity does not exist.

Examples

If a command is executed for a specific existing TransMUX asynchronous port entity to retrieve all parameters but the entity has only near-end (thus no far-end) performance monitoring enabled, then a response will be returned containing only the near-end values.

Restrictions on current bin interval

Current bins can only be retrieved for one slot. The AID values “all” and “usall” cannot be used on slot level to retrieve current bins.

Current bins can only be retrieved if the TransMUX pack is plugged-in. If the retrieval of current bins is not possible because of one of the reasons listed above, then the system does not return a response on the **RTRV-TMUXPM** command.

Example input/output

The following example shows the successful completion of a **RTRV-TMUXPM** command by the network element for a TMAT3 port:

```
RTRV-TMUXPM-TMAT3:UNITE-NE:1-1-#-#-39-tma16:123456::\
cvp-egr,1-up,,,15-min ,,00-00&&08-00;

   UNITE-NE 00-01-01 08:00:00
M  123456 COMPLD
   ″1-1-#-#-39-tma16: cvp-egr,17,,tca,,,15-min,,00-00″
   ″1-1-#-#-39-tma16: cvp-egr,10,,tca,,,15-min,,00-15″
   . . .
   ″1-1-#-#-39-tma16: cvp-egr,15,,tca,,,15-min,,08-00″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-TMUXPM** command.

Related TL1 messages

- “RTRV-PM: Retrieve performance monitoring” (p. 3-2731)
- “RTRV-EPM: Ethernet performance monitoring” (p. 3-2352)
- “RTRV-LOXCPM: Retrieve performance monitoring data of LOXC pack” (p. 3-2500)
- “DLT-TCA-PROF: Delete TCA profile” (p. 3-259)

- "ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
- "ENT-TCA-PROF: Enter TCA Profile" (p. 3-1894)
- "RTRV-TCA-ASGNMT: Retrieve TCA assignment" (p. 3-2979)
- "RTRV-TCA-PROF: Retrieve TCA profile" (p. 3-2982)

RTRV-ULS: Retrieve upper layer stack

The **RTRV-ULS** command is used to retrieve global parameters of the TCP/IP stack.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ULS**:tid*::ctag;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the “RTRV-HDR: Retrieve header” (p. 3-2417) command for input parameter syntax and description of this parameter.

Output format

If the RTRV-ULS request is successful, the following normal completion message is returned:

```
   sid date time
M  ctag COMPLD
   ″spec_block″
   .    .    .
   .    .    .
   .    .    .
   ″spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

1. **`ip_ad_astn`** – IP Address of ASTN
   This parameter is used to set or retrieve the router ID for the SCN and the ONNS node.
   Note: As a consequence, if ip_ad_astn is not configured the whole SWISS application can not be enabled. See also parameter xcsetup in RTRV/ED-EQPT (system).
   Type: Four dot separated numeric strings OR empty string.
   The empty string value is used as indication for an undefined IP address.
   Range: Each numeric string: 0 .. 255.
   Default: empty string.
2. **`ip_ad_uni`** – IP address for UNI
   This parameter is used to retrieve the UNI node IP address.
   Type: Four dot separated numeric strings OR empty string The empty string value is used as indication for an undefined IP address.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.

   The values 0.0.0.0 and 1.1.1.1 are not allowed.
   Default: empty string.
3. **`sigPCIP_Address`** – IP address for Signaling Controller (E-NNI node)
   This parameter is used to retrieve the signaling controller (E-NNI node) IP address.
   Type: Four dot separated numeric strings OR empty string The empty string value is used as indication for an undefined IP address.
   Range:
   - For first numeric string: 0 .. 223
   - For second to fourth numeric string: 0 .. 255.

   The values 0.0.0.0, 1.1.1.1, and 127.0.0.1 are not allowed.
   Default: empty string.
4. **`sigPCID`** – ID of the Signaling Controller (E-NNI node)
   This parameter is used to retrieve the signaling controller (E-NNI node) ID.
   Type: Four dot separated numeric strings OR empty string The empty string value is used as indication for an undefined IP address.
   Range: Four times numeric string: 0 .. 255
   Default: empty string.
5. **`ip_drad`** – IP Default Router Address
   This parameter is used to set or retrieve the IP default router address (IPv4) to which the host points to, needed for the T-TD functionality.
   Type: Four dot separated numeric strings OR empty string. The empty string value is used as indication for an invalid/undefined IP address.
   Range: Each numeric string: 0 .. 255.
   Default: empty string.

6. **`ip_drlan`** – IP Default Router LAN port AID
   This parameter specifies with which LAN port the default router is associated.
   Type: AID.
   Range: LanPortAID. The value lanall is not allowed.
   Default: 1-1-lan1.
7. **`lv_port`** – T-TD Length Value Port
   This parameter is used to set or retrieve the port number over which the T-TD function is accessible using the Length Value encoding method.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3081.
8. **`raw_port`** – T-TD Raw Mode Port
   This parameter is used to set or retrieve the port AID of number over which the T-TD function is accessible using raw-mode encoding.
   Type: Numeric string.
   Range: Each numeric string: 0 .. 65535, except the value 3083.
   Default: 3082.

Detailed behavior description

The command retrieves system parameters as described in the Output parameter section.

Example input/output

The following example shows the result of a RTRV-ULS command:

```
RTRV-ULS:LT-UNITE::123456;
   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
   ”ip_ad_astn=127.93.94.10”
   ”ip_ad_uni=221.131.82.1”
   ”ip_drad=198.84.65.20”
   ”ip_drlan=1-1-lan1”
   ”lv_port=3081”
   ”raw_port=3082”
;
```

The following example shows the result of a RTRV-ULS command:

```
RTRV-ULS:LT-UNITE::123456;
   LT-UNITE 06-06-30 08:00:00
M  123456 COMPLD
   ”ip_ad_astn=127.93.94.10”
   ”ip_ad_uni=221.131.82.1”
   ”ip_drad=198.84.65.20”
   ”ip_drlan=1-1-lan1”
   ”lv_port=3081”
```

```
   "raw_port=3082"
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-NE** command.

Related TL1 Messages

“ENT-ULS: Enter upper layer stack” (p. 3-1944)

RTRV-ULSDCC-L3: Retrieve layer 3 related parameters

The command **RTRV-ULSDCC-L3** is used to retrieve the layer 3 parameters of the OSI nodes or the SCN node. Up to 8 independent OSI nodes can exist in an NE: Node-1 has an NSAP and is created by default. Node 2-8 have each a Network Entity Title (NET). NSAP = NET + SEL . Node 2-8 can be created and deleted, see commands **ENT-ULSDCC-L3**, **DLT-ULSDCC-L3**. The SCN node is created when an ASTN IP address is defined (see **ENT-ULS**), for this node the command retrieves only the NSAP parameter used by I-ISIS. For OSI nodes the parameters of the NSAP field, IS level, IS domain, RA prefix and RA destination node are retrieved.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ULSDCC-L3**:*tid::ctag[::osinode];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack and one SCN node. Node-1 is default. Node 2-8 can be created and configured, see command **ENT-ULSDCC-L3**. The SCN node is created when an ASTN IP address is defined, see ENT-ULS.
   Type AID.
   Range: OSI_node_AID | SCN_node_AID.
   Default: If this field is omitted then the command shall provide output for all nodes which are created.

Output format

If the **RTRV-ULSDCC-L3** is successful, the following normal completion response is returned:

```
sid date time
```

```
M  ctag COMPLD
   "osinode:spec_block"
        ...
   "osinode:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **osinode**
   See definition of input parameter *osinode* in **RTRV-ULSDCC-L3**. The value *osinodeall* is not allowed as output parameter.

Spec block parameters

1. **autoencap** – Auto Encapsulation
   The parameter *autoencap* is used to enable or disable the Auto Encapsulation function. The parameter is only supported for osinode1. The parameter shall be omitted for all other osinodes.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: DISABLE.
2. **l3afi** – NSAP/NET Authority and Format Identifier
   This parameter sets or retrieves the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 10 – 99
   Default: 39. This value identifies the iso_dcc format.
3. **l3idi** – NSAP/NET initial Domain Identifier
   This parameter consists of two values:
   - a 3 semi-octet (3-digit hexadecimal) IDI part to specify the country code indicating the registration authority responsible for the allocation and assignment of the NSAP address.
   - a 1 semi-octet (1-digit hexadecimal) IDI pad part. The parameter is used for set and retrieve values.

   Type: Hexadecimal number.
   Range: 0000 – FFFF
   Default:
   SONET: IDI = 840; IDI pad = F
   SDH: IDI = 000; IDI pad = 0.

4. **`l3dfi`** – NSAP/NET DSP Format Identifier
   This parameter (2-digit hex) NSAP is used to set or retrieve the format for the rest of the address.
   Type: Hexadecimal number.
   Range: 00 – FF.
   Default: 80. This value represents the SONET DSP format.
5. **`l3org`** – NSAP/NET Organization Identifier
   This parameter of NSAP is used to set or retrieve the NSP (allocated Company Code).
   Type: Hexadecimal number.
   Range: 000000 – FFFFFF.
   Default: 000000.
6. **`l3res`** – NSAP/NET Reserved
   This NSAP part currently has not been assigned a specific purpose.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
7. **`l3rd`** – NSAP/NET Routing Domain
   This parameter of NSAP is used to set or retrieve a unique routing domain within an administrative domain. A standard for use has not been defined until now.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
8. **`l3area`** – NSAP/NET Area Identifier
   This parameter of NSAP is used to set or retrieve the area. So NEs can be identified in the same area.
   Type: Hexadecimal number.
   Range: 0000 – FFFF.
   Default: 0000.
9. **`l3sys`** – NSAP/NET System Identifier
   This system parameter of the NSAP is set by the Equipment Manufacturer. The value is fix and globally unique among all NEs. For values of NSAP/NET system parameter see chapter Default below.
   Type: Hexadecimal number.
   Range: 000000000000 – FFFFFFFFFFFF
   Default:
   - Node-1: IEEE-MAC address of LAN1
   - Node-2: IEEE-MAC address of LAN1 +1
   - Node-3: IEEE-MAC address of LAN1 +2
   - Node-4: IEEE-MAC address of LAN1 +3
   - Node-5: IEEE-MAC address of LAN1 +4
   - Node-6: IEEE-MAC address of LAN1 +5

- Node-7: IEEE-MAC address of LAN1 +6
- Node-8: IEEE-MAC address of LAN1 +7
- scnnode: IEEE-MAC address of LAN1 +8 (default)

10. **`l3sel`** – NSAP Selector
    This parameter serves to differentiate multiple access points (NSAPs) for the same NE (node-1). The values cannot be set manually. The NET addresses of node 2-8 have *l3sel*=00.
    Type: Hexadecimal number.
    Range: 00 – FF. Values:
    - 1D: For TP4 PDUs, whenTP4 runs over CLNP, Also used for SDH supporting NEs
    - 00: For NET, used for exchanging PDUs in the ES-IS and IS-IS routing exchange protocols
    - 04: For NSAP of CLNP PDUs, encapsulated IP packets
    - AF: For TARP PDUs, when TARP is run over CLNP.

    Default: 1D, see NET addresses of node 2-8 are reported as NSAP with *l3sel*=00.
11. **`l3isisnl`** – Layer 3 IS Level
    This parameter indicates the network router type: ES-IS (level1, value L1), IS-IS (level2, value L2) and partition repair capability (value3).
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Alphanumeric string.
    Range:
    - L1: this means level1, IS-IS, only allowed when parameter l3radesnode=0
    - L12: this means level2, IS-IS which contains also level1, ES-IS
    - L12PRC : this means partition repair capable (PRC) and contains also level1 and level2.

    Default: L1.
12. **`l3domain`** – Layer3 IS Domain
    Per node in a NE, a domain group must be specified. This will be done with this parameter which can be set and retrieved. All nodes in the NE with the same domain-number are interconnected and can communicate without the need for inter-domain routing via RA-PREFIX.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Alphanumeric string.
    Range:
    - 1 for osinode1
    - 1 – 2 for osinode2 to osinode8.

    Default: 1.

13. **`l3raprefix`** – Layer3 Reachable Address Prefix
    This value specifies an area address or a part of an area address representing an area address group. This parameter is needed when nodes in different routing domains (see *l3domain*) must communicate. The node must forward then CLNP PDUs to the RA-Destination-Node if the destination NSAPs of CLNP PDUs match with the PA-Prefix.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: Hexadecimal number.

    Range:

    - NONE,
    - a string of 1 to 13 bytes: first byte (most significant): 10 – 99; rest: 0000 – FFFF.

    The value represents an area address or a part of an area address beginning with AFI.
    Default: empty string.
14. **`l3radesnode`** – Layer3 Reachable Address Destination Node
    This parameter is in combination with parameter l3raprefix necessary. The value represents the destination node in the same NE. An internal connection over a domain-boundary is established between this node and the RA-Destination-Node, with IS-IS disabled over that connection.
    Note: This parameter is not reported when output parameter *osinode* = scnnode.
    Type: AID and digit value.

    Range:

    - 0, osinode{2-8} for osinode1
    - 0, osinode1 for osinode2 to osinode 8.

    Value 0 indicates that functionality of RA-Prefix is disabled.
    Default: 0.
15. **`nsapcount`** – Number of OSI Nodes in level 1 Area
    This parameter indicates the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the osinode parameter is located. When the number of OSI nodes in the level 1 area is more than the maximum supported number (250), the value of this parameter is shown as ″250+″,which indicates a

state of NSAP count overflowed. The parameter is omitted for the SCN node.
Type: Value set.
Range: 1 – 250 if the number of nodes is less or equal to 250. 250+ if the number of nodes exceeds 250.
Default: 1

16. **`nsapthlev`** – Number of OSI Nodes Threshold Level
This parameter indicates the NSAP count threshold level for monitoring the number of OSI nodes (NSAPs) for the level 1 area in which the OSI node specified by the *osinode* parameter is located. A threshold value of zero (0) indicates that NSAP threshold alarming is disabled. The parameter is omitted for the SCN node.
Type: Numeric string.
Range: 0 no alarming; 1 – 250 threshold alarming
Default: 0

Detailed behavior description

The command retrieves system parameter as described in the Output parameter section.

Example input/output

The following example shows the result of RTRV-ULSDCC-L3:

```
RTRV-ULSDCC-L3:NE-UNITE::123456::osinode1;
   NE-UNITE 00-12-24 11:01:59
M  123456 COMPLD
   ″osinode1:L3AFI=39,L3IDI=840F,L3DFI=80,
   L3ORG=100100,L3res=0000,L3RD=0000,L3AREA=0000,
   L3SYS=08006A112345,L3SEL=1D,L3ISISNL=L12PRC,
   L3DOMAIN=1,L3RADESNODE=0,NSAPCOUNT=2,NSAPTHLEV=0″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the RTRV-ULSDCC-L3 command.

Related TL1 messages

- “ENT-ULSDCC-L3: Create a further OSI node” (p. 3-1951)
- “ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack” (p. 3-1312)
- “DLT-ULSDCC-L3: Delete a node ID other than node-1” (p. 3-265)

RTRV-ULSDCC-L4: Retrieve OSI layer 4 related parameters

The command **RTRV-ULSDCC-L4** is used to retrieve layer 4 parameters of the OSI stacks. Layer 4 include some TARP parameter which can be modified and the TARP Manual Adjacencies parameter which can be provided, modified or deleted.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-ULSDCC-L4**:*tid::ctag[::osinode];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
2. **ctag** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for input parameter syntax and description of this parameter.
3. **osinode**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack and one SCN node. Node-1 is default. Node 2-8 can be created and configured, see command **ENT-ULSDCC-L3**. The SCN node is created when an ASTN IP address is defined, see ENT-ULS.
   Type AID.
   Range: OSI_node_AID | SCN_node_AID.
   Default: If this field is omitted then the command shall provide output for all nodes which are created.

Output format

If the RTRV-ULSDCC-L4 is successful, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   "osinode:spec_block"
        ...
   "osinode:spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Manual adjacencies

If more than one manual adjacency is defined for one osinode, they appears on separate output lines.

Example:

```
   sid date time
M  ctag COMPLD
   ″osinode1:<other spec_block parameters>″
   ″osinode1:l4ajnsap=39840F800000000000000000123456789ABC00″
   ″osinode1:l4ajnsap=39840F800000000000000000023456789ABCD00″
;
```

Output parameters

Command parameters

1. **`osinode`**
   A 1675 LambdaUnite MSS NE can have up to 8 instances of the OSI stack and one SCN node. Node-1 is default. Node 2-8 can be created and configured, see command **`ENT-ULSDCC-L3`**. The SCN node is created when an ASTN IP address is defined, see ENT-ULS.
   Type AID.
   Range: OSI_node_AID | SCN_node_AID.
   Default: If this field is omitted then the command shall provide output for all nodes which are created.

Spec block parameters

1. **`l4etdc`** – TARP Data Cache
   This parameter is only valid for node-1. It is used to enable or disable TARP data cache.
   Type: Alphanumeric value set.
   Range: ENABLE (default), DISABLE.
2. **`l4etof`** – TARP origination
   This parameter only valid for node-1. It is used to enable or disable TARP Origination function and is valid for all interfaces of a node.
   Type: Alphanumeric value set.
   Range: ENABLE, DISABLE.
   Default: ENABLE (for osinode1), DISABLE (for osinode2-8, fixed).

3. **`l4etpf`** – TARP propagation
   This parameter is used to enable or disable TARP Propagation function for a node.
   Type: Alphanumeric value set.
   Range:
   - ENABLE (default)
   - DISABLE.
4. **`l4etrf`** – TARP responder
   This parameter is only valid for node-1. It is used to enable or disable TARP responder function
   Type: Alphanumeric value set.
   Range:
   - ENABLE
   - DISABLE.

   Default: For node-1: ENABLE DISABLE (fixed for nodes 2-8).
5. **`l4ajnsap`** – TARP Manual Adjacency List
   This parameter specifies the NSAP address portion of an entry in the TARP Manual Adjacent NE list. Up to two addresses are possible for each node in a 1675 LambdaUnite MSS NE.
   Type: Hexadecimal number or alphanumeric value set.
   Range: NONE, 20 octet (40-digit hex) long address.
   The NSAP address / NET is composed of the following components:
   - l4ajafi 10 – 99
   - l4ajidi 0000 – FFFF
   - l4dfi 00 – FF
   - l4alorg 000000 – FFFFFF
   - l4ajres 0000 – FFFF
   - l4jrd 0000 – FFFF
   - l4jarea 0000 – FFFF
   - l4ajsys 000000000000 – FFFFFFFFFFFF
   - SEL 00 The last byte indicates the selector and is not to set.

   Note: For more details see description of ENT-ULSDCC-L3.
   Default: NONE (no TARP manual adjacencies).

Detailed behavior description

The command retrieves system parameter as described in the Output parameter section.

Example input/output

The following example shows the result of RTRV-ULSDCC-L4:

```
RTRV-ULSDCC-L4:NE-UNITE::456789::osinode1;
```

```
   NE-UNITE 01-17-01 18:11:01
M  456789 COMPLD
   "OSINODE1:l4etof=DISABLE,l4etpf=ENABLE,l4etrf=ENABLE,
   l4etdc=ENABLE,
   l4ajnsap=39840F80000000000000000000123456789ABC00"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-ULSDCC-L4 command.

Related TL1 messages

- "ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
- "DLT-ULSDCC-L4: Delete TARP manual adjacent NSAP address in layer 4 of the specified OSI stack" (p. 3-269)

RTRV-USER: Retrieve user

The **RTRV-USER** command retrieves a user's own security information.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S2

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-USER**:*tid*::*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output format

After receiving the **RTRV-USER** command, the following normal completion response is returned:

```
^^^sid^date^time
M^^ctag^COMPLD
^^^"uid::spec_block"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **uid** – User ID
   User ID whose security parameters are to be retrieved.
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of uid values for the **ACT-USER** command are not allowed. Valid uid values for the **ACT-USER** command are case-sensitive

alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
Type: Alphanumeric string (case-sensitive)
Range: 5 – 10 characters. The uid ALL is not allowed in any combination of upper or lower cases.

Spec block parameters

1. **`page`** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90

2. **`screen`** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string An '&' is also allowed for concatenation of values.
   Range:
   - DBCHG
     Data Base Change notifications – changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes). The user will receive **`REPT DBCHG`** messages.
   - STCHG
     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **`REPT DBCHG`** messages.
   - PSCHG
     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS
     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
   - ONNS
     ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
   - OWN
     Responses to own commands. The user will receive no other notifications.
     Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.
   - NA

Not applicable. This value is reserved for users with uidclass=MEMADMIN. The notification behavior is controlled in this case by the user class.

Default: OWN
If more than one value is set the values will be concatenated by ’&’.
Example: DBCHG&STCHG&ALARMS

3. **`tmout`** – Inactivity Timeout Period
   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.
   Type: Integer.
   Range: 0 – 999 [minutes]
   Default: 30

4. **`uidclass`** – UID Class
   This is the user class of a *uid*.
   Type: Alphanumeric value set.
   Range:
   - OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
   - MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **`RTRV-AO`**.

   Default: OTHER

5. **`upc`** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands (’&’). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation “&” is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also “List of user privilege codes” (p. 15-1).

Example input/output

The following example shows the successful completion of **`RTRV-USER`** command by the NE:

`RTRV-USER:LT-UNITE-123::123456;`

```
   sid date time
M  ctag DENY
```

```
   SDNC
   /* Status, Data Not Consistent,invalid instance of entity */
;
```

Related TL1 messages

- "ED-USER: Edit user" (p. 3-1323)
- "RTRV-USER-SECU: Retrieve user security" (p. 3-3063)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

RTRV-USER-SECU: Retrieve user security

The **RTRV-USER-SECU** command can be initiated by a user to retrieve user security information of one or all the users in the NE.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-USER-SECU**:*tid*:[*uid*]:*ctag*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **uid** – User ID
   If this parameter is omitted, all *uid*s will be retrieved.
   See definition of “ACT-USER: Activate user” (p. 3-24).
3. **ctag** – Correlation Tag
   See definition of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output format

If the NE has logins to report, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ″uid::spec_block″
   .
   .
   ″uid::spec_block″
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **uid** – User ID
   This is the unique user login identifier for which the ACT-USER login command is intended. Grouping and/or ranging of uid values for the **ACT-USER** command are not allowed. Valid uid values for the **ACT-USER** command are case-sensitive alphanumeric strings which have been previously provisioned as valid login IDs in the NE.
   Type: Alphanumeric string (case-sensitive)
   Range: 5 – 10 characters. The uid ALL is not allowed in any combination of upper or lower cases.

Spec block parameters

1. **alw_login** – Allow Login
   This parameter displays the enable/disable state of a User ID.
   A User ID can be automatically disabled by the system when the time specified by parameter *usrage* (see “RTRV-NE-SECU: Retrieve NE security” (p. 3-2552)) expires or via the **ED-USER-SECU** command. A User ID can be enabled via the **ED-USER-SECU** command.
   If a User ID is disabled, logins with that User ID are denied.
   Type: Alphanumeric value set.
   Range:
   - YES: User ID is enabled
   - NO: User ID is disabled

   Default: YES
2. **initobs** – Initialization Observability
   This parameter allows the system to accept commands before it has completed initialization. Do not enable this parameter in networks using a Telcordia operations system.
   Type: Alphanumeric value set.
   Range: DISABLE – The feature is disabled.
   Default: DISABLE
   Note: 1675 LambdaUnite MSS does not support Initialization Observability feature. This parameter is only for compatibility. The DISABLE value will be returned as fixed value.
3. **page** – Password Aging Interval
   This parameter specifies the period in days after which the user has to change the password of their account. If the value is set to 0 then the password aging function is disabled.
   Type: Integer.
   Range: 0, 7-999 [days]
   Default: 90

4. **`screen`** – Message Screening
   This parameter specifies what output messages are associated with the *uid*. This parameter is omitted if the user has none of these values. Any combination of these values may be specified using ampersand (&)
   Type: Alphanumeric string An ’&’ is also allowed for concatenation of values.
   Range:
   - DBCHG
     Data Base Change notifications: changes in creation of entities, in user provisionable parameters, in deletion of entities will be reported (data base changes).
     File Transfer In-progress notifications: The user will receive **`REPT DBCHG`** and REPT EVT FXFR messages.
   - STCHG
     State Change notifications – report changes in user non-provisionable parameters (state changes). The user will receive **`REPT DBCHG`** messages.
   - PSCHG
     Protection switch notifications – report protection switch changes. User will receive REPT EVT and REPT SW messages.
   - ALARMS
     Alarm notifications. The user will receive REPT ALM ENV, REPT ALM, and REPT EVT messages.
   - ONNS
     ONNS Notifications. Database and state changes associated with entities managed by ONNS application and only issued when *xcsetup* = NN. The related ONNS TL1 notifications are REPT PTHCHG, REPT PTHEVT, REPT NNCHG, REPT NNEVT.
   - SECURITY: Security event notifications
   - OWN
     Responses to own commands. The user will receive no other notifications.
     Note: OWN will never be shown in conjunction with DBCHG, STCHG, PSCHG or ALARMS.
   - NA
     Not applicable. For users with uidclass=MEMADMIN only this value is allowed. The notification behavior is controlled in this case by the user class. If the value NA is applied for users of other user classes, then the value has the same effect as DBCHG.

   Default: OWN
   If more than one value is set the values will be concatenated by ’&’.
   Example: DBCHG&STCHG&ALARMS

5. **tmout** – Inactivity Timeout Period
   If there are no messages between the user and the NE over the Timeout Period, the session is logged off. A value of 0 disables the timeout function.
   Type: Integer.
   Range: 0 – 999 [minutes]
   Default: 30
6. **ucpl** – User Community Priority Level
   This parameter specifies with which priority the commands of a user will be operated by the NE. Input commands from users with higher priority are executed before commands from users with lower priority.
   Type: Integer.
   Range: 1
   Default: 1
   Note: 1675 LambdaUnite MSS does not support User Community Priority Level's. Only the default value 1 will be reported. The value is not setable.
7. **uidclass** – UID Class
   This is the user class of a *uid*.
   Type: Alphanumeric value set.
   Range:
   - OTHER – This is the default class. For users in this class, command privileges and notifications are controlled only by their UPCs and their notification Registration List.
   - MEMADMIN – User class for memory administration users. All users of this class share the same atag counter for autonomous messages and are allowed to execute the command **RTRV-AO**.

   Default: OTHER
8. **upc** – User Privilege Code List
   This parameter specifies the user privilege code UCFC/UCAL pair assigned to a user. Multiple UCFC/UCALs can be specified using single ampersands ('&'). If a user tries to operate a command with a privilege code higher than assigned to the user for the category of the command the command will be denied.
   Type: Alphanumeric string; for concatenation "&" is also allowed.
   Range: P***i***, M***i***, T***i***, S***i***, PM***i***, D***i***, C***i***, where ***i*** is an integer ranging from 0 to 5, with ***i*** = 0 implying that there is no authorization for the functional category. S0 is not allowed.
   Default: P1&M1&T1&S1&PM1&C0 except for super users: P5&M5&T5&S5&PM5&C5.
   Note: The system does not support commands for the D (DEBUG) category. Setting this privilege code does not have any effect on the system behavior and it will not be visible in output direction. It is only for compatibility.
   See also "List of user privilege codes" (p. 15-1).

Example input/output

Example for specified user:

The following example shows the successful completion of **RTRV-USER-SECU** command by the NE. The example below retrieves the security information associated with the user specified in the *uid* field:

RTRV-USER-SECU:LT-UNITE-123:nbguser1:123456;

```
   sid date time
M  ctag DENY
   IENE
   /* Input, Entity Not Exists */
;
```

Example for all users:

The following example shows the successful completion of **RTRV-USER-SECU** command by the NE. The example below retrieves the security information associated with all the users from the NE:

RTRV-USER-SECU:LT-UNITE-123::123456;

```
   LT-UNITE-123 05-05-31 08:00:00
M  123456 COMPLD
   "LUC01::upc=S5,screen=DBCHG&STCHG&PSCHG&ALARMS,page=90,tmout=30,alw_
      login=YES,uidclass=OTHER"
   "LUC02::upc=S5,screen=DBCHG&STCHG&PSCHG&ALARMS,page=47,tmout=11,alw_
      login=YES,uidclass=OTHER"
   "hugo::upc=P3&M3,screen=DBCHG&STCHG&ALARMS,page=999,tmout=3,alw_
      login=YES,uidclass=OTHER"
   "egon::upc=P3&M3&T3&PM3,screen=PSCHG,page=7,tmout=999,alw_
      login=NO,uidclass=OTHER"
   "ronald::upc=T2,screen=DBCHG&STCHG&PSCHG&ALARMS,page=0,tmout=0,alw_
      login=YES,uidclass=OTHER"
   "fritz::upc=C5,screen=OWN,page=0,tmout=0,alw_login=YES,uidclass=OTHER"
   "cathrin::upc=T2,screen=OWN,page=90,tmout=30,alw_login=YES,uidclass=OTHER"
;
```

Error responses

Validation rule: invalid uid

If the NE receives an RTRV-USER-SECU command with an invalid (unknown) Only the own *uid* value.

Error code: `IDNV – Input, Data Not Valid, invalid UID`

Related TL1 messages

- "ED-USER: Edit user" (p. 3-1323)
- "ENT-USER-SECU: Enter user security" (p. 3-1970)

- "ED-USER-SECU: Edit user security" (p. 3-1328)
- "DLT-USER-SECU: Delete user security" (p. 3-272)
- "RTRV-USER: Retrieve user" (p. 3-3059)

References

TR-NWT-000835, OTGR Section 12.5: Operations Applications Messages – NE and Network System Security Administration Messages.

RTRV-VC3: Retrieve VC3 tributary

General

The command **`RTRV-VC3`** is used to retrieve all parameters related to a VC3 tributary.

Note: If a tributary AID (Access IDentifier) is specifed in the RTRV-VC3 request for a slot provisioned with a GE1 pack (e.g. 1-1-#-#-1-1-1 is used), then an empty reply is returned instead of: `IENE – Input, Entity Not Exists`.

For GE1 the "RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109) command must be used.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **`RTRV-VC3`**:*tid*:*aid*:*ctag*;

Command parameters

1. **`tid`** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.
2. **`aid`** – Access Identifier
   AID of a SDH tributary. The keyword **`all`** is allowed on slot, port and tributary level. The so called ″dynamic parameters″ is reported, if a **`all`** is used at the tributary level.
   The ″dynamic parameters″ are:
   - *ptrc*, *ptrcst*, *siglb*, *trailstate*.
     Type: AID
     Range: STS1NumAID.
3. **`ctag`** – Correlation Tag
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and description of this parameter.

   Important! To retrieve the dynamic parameters (*ptrc*, *ptrcst*, *siglb*, *trailstate*) of all tributaries use the ("RTRV-ALL: Retrieve all (tributary)" (p. 3-2228)) command and set the parameter *dyn* to YES.

Output format

In response to a valid RTRV-VC3 command, the following output report is returned:

```
   sid date time
M  ctag COMPLD
   "aid[:spec_block]"
       ...
       ...
   "aid[:spec_block]"
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   Description: AID of the SDH tributary. The keyword **`all`** is not allowed on any level.
   Type: AID
   Range: STS1NumAID

Spec block parameters

1. **`brstintvl`**– Bursty Interval
   Measurement period for determining bursty errors The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range: 2 to 10.
   Default: 7.
2. **`brstthr`** – Bursty Error Threshold
   The values are set in percent for the degradation threshold. The setting of this parameter is used when *sdsfmode* = BURST.
   Type: numeric value set.
   Range (as percentage): 5, 10, 15, 20, 25, 30, … 50.
   Default: 30.
3. **`crslpbkrate`** – Cross-connect loopback Rate
   It is independent from the current interface standard and therefore both SDH and SONET values are allowed.
   The parameter *crslpbkrate* is not reported if the parameter *crlspbkstat* indicates that no loopback is set.
   Type: Alphanumeric value set.

Range:

- VC3
- VC4
- VC44C
- VC416C
- VC464C
- STS1
- STS3
- STS12
- STS48
- STS192

4. **`crslpbkstat`** – Cross-connect loopback status
   State of cross- connect loopback.
   Type: Alphanumeric value set.
   Range:
   - YES = Cross-connect loopback is provisioned
   - NO (default) = No cross-connect loopback is provisioned.
5. **`degthr`** – Signal Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON
   Type: numeric value set.
   Range (as exponents of 10): –5; –6 (default); –7; –8; –9
6. **`dexcthr`** – Excessive Degrade Threshold
   The setting of this parameter are used when *sdsfmode* = POISSON. The values are set as exponent of 10.
   Type: numeric value set.
   Range (as exponent of 10) –3 (default); –4; –5.
7. **`exphptrc`** – Expected Path Trace (J1)
   The parameter *exphptrc* specifying the expected path trace is only relevant for the path trace supervision when the path trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Hexadecimal digits
   Range: 16 byte string. Each byte is expressed as two hexadecimal digits (example value: 6D). The first byte value is in the range from 80 to FF. The other byte values are in the range from 00 to 7F.
   Default: 89hex followed by 15 bytes 00hex.
8. **`feppm`** – Far-End Path PM Enable
   It indicates whether this performance monitoring feature is enabled or disabled.
   Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

9. **hptimmon** – Path Trace Monitoring Enable (J1)
Description: High order path trace (J1) identifier mismatch (TIM) monitor enable. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.

   Range:

   - ENABLE
   - DISABLE (default).

10. **loca** – Ring A-node TID
This parameter specifies the TID of the node at which the service is added to the BLSR ring. This parameter is required for the input of tributaries to be used by ONNS on ports of *portClass* = EDGE being members in BLSR/MS-SPRing protection groups. The *loca* parameter value is an empty string for all other tributaries.
loca is introduced specifically for use related to ONNS application.
Type: TID or empty-string.
Range: Please refer to the TID definition in "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: "". This is displayed as \"\" in output direction.
11. **locz** – Ring Z-node TID
This parameter specifies the TID of the node at which the service is dropped from the BLSR ring. This parameter is required for the output of tributaries to be used by ONNS on ports of *portClass* = EDG E being members in BLSR/MS-SPRing protection groups. The *locz* parameter value is an empty string for all other tributaries.
locz is introduced specifically for use related to ONNS application.
Type: TID or empty-string.
Range: Please refer to the TID definition in the "RTRV-HDR: Retrieve header" (p. 3-2417) description.
Default: empty string: "". This is displayed as \"\" in output direction.
12. **neppm** – Near-End Path PM Enable
It indicates whether this performance monitoring feature is enabled or disabled.
Type: Alphanumeric value set.

    Range:

    - ENABLE
    - DISABLE (default).

13. **pjepm** – Pointer Justification Event PM Enable
It indicates whether this performance monitoring feature is enabled or disabled.
Type: Alphanumeric value set.

Range:

- ENABLE
- DISABLE (default).

14. **`pmhpfeses`** – Performance Monitoring Path Far-End SES Threshold
This parameter describes the Path Far-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
15. **`pmhpneses`** – Performance Monitoring Path Near-End SES Threshold
This parameter describes the Path Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severely errored. When the interface standard is changed the current default value is not adapted automatically.
Type: numeric value set.
Range (superset of ranges for SDH and SONET): 1-64000.
Default: 2400.
16. **`ptca`** – Path TCA Profile Pointer
Type: Alphanumeric string.
Range: string with up to 24 characters.
Default: DEFAULT0.
17. **`ptrc`** – Path Trace (J1)
This parameter shows the value of the path trace identifier on the incoming path overhead J1 byte of a non-terminated tributary. The length depends on the value of the parameter *ptrcrfmt*. The status of this value is contained in parameter *ptrcst*. If *ptrcst* indicates NORMAL, then ptrc contains a valid value. If *ptrcst* indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to a signal-fail condition or intra-NE communication problems, and the output parameter ptrc is omitted from the response. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
Type: Hexadecimal (each byte is expressed as two hexadecimal digits. An example of a byte is 6D).
Range:
    - 1 byte for non-specific repeating byte
    - 16 byte: These 16 byte shall consist of 15 byte information and the CRC7 byte
    - 64 bytes: This is a pure byte stream without any restrictions.
18. **`ptrcrfmt`** – Path Trace Read Format (J1)
The read format of J1, the incoming path trace (ptrc).
Type: numeric string.

Range:

- 2: for non-specific repeating byte
- 16: for 16 byte specific string.
- 64: for 64 byte string.
  Default: 2.

19. **ptrcst** – Status of Incoming Path Trace (J1)
    This parameter shows the status of the value of the incoming path trace identifier which is available with parameter ptrc. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
    Type: Alphanumeric value set.
    Range:
    - NORMAL: ptrc contains a valid value
    - UNAVAILABLE: no valid value for ptrc could be obtained by the system
    - AIS: no valid value for ptrc due to an SF condition.
20. **sdsfmode** – Signal Degrade/Signal Fail Mode
    The degraded state is detected in BURST or POISSON mode.
    Type: Alphanumeric value set.
    Range:
    - BURST (default)
    - POISSON
21. **siglb** – Signal Label
    This parameter shows the value of the signal label on the incoming path overhead C2 byte of a non-terminated VC tributary. The parameter is not reported if the *trailstate* parameter indicates a fail condition, i.e. has the value AIS or UNAVAILABLE. For concatenated tributaries the system shall not return this parameter (see detailed behavior below) .
    Type: Alphanumeric value set.
    Range: 00, 01, …, FE, FF; i.e. two hexadecimal digits.
22. **substr** – Substructured
    This parameter indicates if the tributary is substructured.
    Substructure for LO cross-connections:
    A HO tributary, which is sub structured for LO cross-connections, has a HO path termination associated which provides LO tributaries to be used for LO cross-connections.
    The parameter *substr* is set autonomously when LO cross-connections are created or deleted with ENT-CRS or DLT-CRS commands (see also the lower order cross-connection creation/deletion behavior description for the commands ENT-CRS and DLT-CRS).
    A sub structured HO tributary cannot be used to set up HO cross-connections. The parameter *substr* is set autonomously when the LO substructure is created or deleted using the **ED-SUBS** command.
    Substructure for conversion cross-connections:

A HO tributary, which is sub structured for the LO side of a conversion cross-connection, has a HO path termination associated which provides LO tributaries to be used for conversion cross-connections.
The parameter *substr* is set autonomously when an AU3/AU4 conversion cross-connection is created.
A HO tributary, which is sub structured for conversion cross-connections, cannot be used to set up HO cross-connections.
Type: Alphanumeric value set.

Range:

- YES: Automatically sub structured for LO cross-connections
- MANSUB: Manually sub structured for LO cross-connections
- CNVSUB: Automatically sub structured for the LO side of conversion cross-connections
- NO (default): Not sub structured, not used for conversion cross-connections

23. **`tcamd`** – TCA mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable:
   - TR-only (also known as the implicit reset method)
   - TR-RTR (also known as the explicit reset method).
   Type: Alphanumeric value set.
   Range: TR, TRRTR (default).
24. **`tmonmd`** – Tributary Monitoring Mode
This parameter indicates the mode of the tributary.
Type: Alphanumeric value set.
Range:
   - MON = alarm monitoring is taking place on the port.
   - NMON (default) = alarm monitoring is not occurring on the port.
25. **`trailstate`** – Trail State (C2)
It is possible to read the accepted trail value status (C2) of every VC which is terminated/monitored in the system. For concatenated tributaries the system shall not return this parameter (see detailed behavior below).
Type: Alphanumeric value set.

Range:

- NORMAL: for accepted TSL which has valid payload type
- AIS: for SSF, VC-AIS, dTIM or dUNEQ
- UNAVAILABLE: for TSL which does not meet the acceptance criteria or no reporting is possible.

Default: UNAVAILABLE.

26. **`tribsdhpn`** – Tributary Alarm Severity Assignment Profile
This parameter points to a tributary ASAP profile name of type PATH for the designated AID.
Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
Range: string with a maximum of 24 characters.
Default: DEFAULT.

Detailed behavior description

Applicability for VC4 and concatenated signals

The command **`RTRV-VC3`** is also applicable to VC4 tributaries. Even though a VC4 signal is not a concatenated VC3 signal in a strict sense, from a pointer processor viewpoint it can be considered to be comprised of three VC3 signals which have the properties of a concatenated signal. Thus it is not necessary to have a separate RTRV command for VC4.

Retrieval of concatenated tributaries

It is allowed to retrieve data from concatenated tributaries and the system responds by returning all parameters which are not derived from traffic. The details can be found in the parameter description.

Concatenated tributaries are such tributaries, that are not the leftmost ones in concatenated signals, e.g. if an OC 48 port is provisioned for a fixed traffic of AU4-16c, STS-1 #1 represents the VC4-16c defined and all other STS-1s #2 to #48 do currently not represent valid substructures and thus are concatenated tributaries. The retrievable values e.g. for the STS-1s #2 to #48 are irrelevant for transmission.

Parameters derived from traffic are data like trace identifiers and signal labels which are collected from the actual input signal and thus do not exist for concatenated tributaries.

Behavior for non-existing entities

If the specified VC3/VC4 tributaries do not exist, the command shall return an empty response. E.g. if an attempt is made, to retrieve VC3/VC4 tributaries on a GbE circuit pack.

Example input/output

```
RTRV-VC3:LT-UNITE:1-1-#-#-3-1-1:123456;

   LT-UNITE 00-11-27 08:00:00
M  123456 COMPLD
   "1-1-#-#-3-1-1:crslpbkstat=NO,tmonmd=NMON,tribsdhpn=Default,
ptca=Default0,neppm=DISABLE,pmhpneses=30,pmhpfeses=30,degthr=-
      6,...,ptrcrfmt=1,dexcthr=-3,ptrc=00,ptrcst=NORMAL,sdsfmode=POISSON"
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the RTRV-VC3 command.

Check CCB against istd for related port

Validation rule: Check CCB against *istd*

Check wether the CCB fits to *istd* of the related port.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`.

E.g. DENY, if a command with rr=VC3 is made on a port which is provisioned as *istd* = SONET.

References

GR-199-CORE (RTRV-*rr*)

RTRV-VCG: Retrieve virtual concatenation group

The **RTRV-VCG** command retrieves the attributes of a virtual concatenation group.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-VCG***[-modifier]:tid:aid:ctag[:::spec_block];*

Input parameters

Command parameters

1. **modifier**
   The modifier specifies the type of the virtual concatenation group of the unit. Implicit, it indicates the transmission rate for which VCG related data is retrieved. The parameter modifier must be given if a virtual concatenation group on a GE10PL1 or ES64U pack is addressed. Otherwise this parameter has to be omitted.
   Type: Alphanumeric value set.
   Range: 1GEPL representing a 1 GE port on the GE10PL1 pack; 10GEPL representing a 10 GE port on the GE10PL1 pack, ES64U representing a VCG port at an ES64U subsystem.
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   This parameter specifies the virtual concatenation group (VCG) to be operated on.
   Type: AID.

   Range: VCGAID as defined in AID structure. The values

   - *vall* for Simple_VCGAID and
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID

- *all* for Simple_ShelfAID and
- *all* for BayAID

are accepted.

4. **`ctag`** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Spec block parameter

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values *usall* or *vall* at the appropriate positions of the input AID. For input AIDs not using *usall* or *vall* this parameter is not allowed. This implies that with such input AIDs the dynamic parameters are always returned.
   The following output parameters of a vcgport port are affected: *mfiv*, *macaddr*, *stmd*, *stpop*, *stpid*, *stprole*, *vss*, *vvidlsd*.
   Note that not only those parameters have to be returned which are supported by the related Ethernet pack. Not all Ethernet packs support all listed parameters. For details refer to the detailed specification of those output parameters.
   For *modifier* ES64U, the parameter *dyn* is not supported because none of these parameters will be returned.
   Type: Alphanumeric value set.
   Range:
   - YES: Report dynamic parameters in the reply as well
   - NO (default): Do not report dynamic parameters in the reply.

Output format

If the NE fully complies with the **`RTRV-VCG`** request, the following normal completion response is returned:

```
sid date time
M  ctag COMPLD
   ″aid::spec_block″
      <1 or more of the line above>
   ″tribaid:mfiv,MFIV″
    <no such line, if the virtual concatenation mode is disabled, else
     no such line, if no dynamic data are retrieved, else
     no such line, if modifier = ES64U, else
     one or more lines, one for each tribaid belonging to aid>
       ...
       ...
   ″aid::spec_block″
```

```
      <1 or more of the line above>
   ”tribaid:mfiv,MFIV”
    <no such line, if the virtual concatenation mode is disabled, else
     no such line, if no dynamic data are retrieved, else
     no such line, if modifier = ES64U, else
     one or more lines, one for each tribaid belonging to aid>
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure. The values
   - *vall* for Simple_VCGAID and
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   are excluded.
2. **`tribaid`** – Tributary Access Identifier
   The tributary access identifier specifies a tributary belonging to the virtual concatenation group (VCG) given by the aid above.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   This specification is necessary due to hardware restrictions of the Gauss device. If the parameter *tribaid* is not supported, then the parameter shall be omitted.
   Type: AID.
   Range: VCGNumAID as defined in AID structure. The values
   - *all* for Simple_VCGNumAID,
   - *vall* for Simple_VCGAID,
   - *usall* for Simple_UnivSlotAID
   - *all* for Simple_UnivSlotAID
   - *all* for Simple_ShelfAID and
   - *all* for BayAID

   are excluded.

3. **mfiv** – Multiframe Indicator Value
   This parameter reports the differential delay between the tributary addressed by *tribaid* (see above) and the fastest tributary of the VCG. The fastest tributary has the value zero. For each retrieve of a VCG there is a report of *mfiv* for all tributaries belonging to the VCG.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   This specification is necessary due to hardware restrictions of the Gauss device. If the parameter *mfiv* is not supported, then the parameter shall be omitted.
   Type: Alphanumeric value set.
   Range:
   - 0, 125, 250, 375, …, 511750, 511875 [microseconds]
   - NA
   - UNAVAILABLE

   In fact the reported range for delay is 0 … 255875 microseconds. Other values can only be reported as raw values without a delay calculation between the tributaries. UNAVAILABLE is reported when the accepted MFI contents do not meet the basic acceptance criteria (multiframe alignment not completed), or intra-NE communication with the reporting entity is disrupted. NA (Not Applicable) is reported under defect conditions detected by the Sn[i] termination function or the VC-n is not part of an S-n-Xv group.

Spec block parameters

1. **clientNodeID** – Client Node IDentifier
   The *clientNodeID* identifies the IP address of a client node attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
   For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
   This attribute replaces attribute “uniClientNodeID”, as a more generic name seems reasonable for a possible extension of this attribute to other eport classes in future.
   This parameter is supported by ports of *eportClass* UNI.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: IPv4 format: Four dot separated numeric string
   Range: Each numeric string: 0…255
   Default: 0.0.0.0
2. **cnprole** – Customer/Network Port Role
   This parameter specifies whether the addressed port is used as a customer WAN port or a network WAN port.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.

If the parameter *cnprole* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: CUSTOMER, for a customer WAN port. NETWORK, for a network WAN port.
Default: NETWORK.

3. **`dftpri`** – Default User Priority
   This parameter specifies the default user priority acc. the IEEE 802.1Q tagging scheme.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   If the parameter *dftpri* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range:
   - HIGH
   - LOW (default).
4. **`dftvlan`** – Default VLAN Identifier
   The parameter *dftvlan* specifies the value of the VLAN identifier which is applied to ingressing frames (tagging) . This only applies to frames carrying no VLAN identifier, yet. This parameter is also used to provision the customer tag (CID) to be used in transparent tagging mode.
   This parameter is not supported if the input parameter *modifier* = 1GEPL, or *modifier* = 10GEPL.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   If the parameter *dftvlan* is not supported, then the parameter is omitted.
   Type: Alphanumeric value set.
   Range: 1, 2, 3, … 4092, 4093; UNAVAILABLE (default)
5. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Ethernet port and VCG port have the same ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
6. **`enettca`** – Ethernet TCA Profile Name
   This parameter specifies the name of the threshold crossing alert (TCA) profile used for the VCG.
   This parameter is not supported if the input parameter *modifier* = ES64U.
   If the parameter *enettca* is not supported, then the parameter is omitted.
   Type: Alphanumeric string.
   Range: 1 … 24 characters length.
   Default: DEFAULT0.

7. **eportClass** – Ethernet Port Class
   This identifies the Ethernet port class of the AID. *eportClass* is used to specify Ethernet ports for use by NN application.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: Alphanumeric value set.
   Range: Depends on the *modifier*:
   - if *modifier* is missing or is 1GEPL, then the value range is: TRADITIONAL, EDGE, UNI
   - if *modifier* is 10GEPL, then the value range is: TRADITIONAL, EDGE
   - if *modifier* is ES64U, then the value range is: TRADITIONAL

   Default: Depends on system parameters Cross Connect Setup (*xcsetup*) and Port Class Default (*portclassdefault*):
   - if *xcsetup* is TRADITIONAL, then the default value is TRADITIONAL
   - if *xcsetup* is SWISS, then the value of *portclassdefault* is used as default value.
   - if *modifier* is ES64U, then the default value is TRADITIONAL.
8. **eportControl** – Ethernet Port Control (Ethernet Port Administrative Mode)
   This identifies the administrative mode of Ethernet ports of *eportlass* = EDGE ot UNI. For Ethernet ports of other Ethernet port class this parameter is not relevant. *eportControl* is introduced specifically for use by NN application.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: Alphanumeric value set.
   Range: NORMAL (default), MAINT, LOCKOUT .
9. **htmr** – Hold-Off Time
   The parameter *htmr* specifies the hold off time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it goes to state faulty then this tributary is taken out of transmission by LCAS after the hold off time. This time is configured on group level, i.e. all tributaries have the same value for hold-off time. This parameter only affects NE behavior if *lcasmd* = ENABLE.
   For *modifier* = ES64U, the parameter is visible with default values for retrieval, but it cannot be modified.
   Type: numeric value set.
   Range: 0 (default), 100, 200, …, 9900, 10000 [milliseconds].
10. **inputsig** – Tributary Input Signal Rate List
   This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure depends on interface standard specified for this VCG by istd . For definition of *istd* refer to “RTRV-VCG: Retrieve virtual concatenation group”

(p. 3-3078) Output parameters.

If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the possible tributary structures depends on interface standard specified for this VCG by istd and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to RTRV-VCG Output Parameters.

Type: numeric string of the form NX

- N specifies the count of container type X
- X specifies the container type (or envelope).

Range:

- Values: N = 7, X = 4; Modifier=omitted; Interface Standard=SDH; Virtual Concatenation=not applicable; Represented Signal Structure: 7 times VC4
- Values: N = 7, X = 4; Modifier=1GEPL; Interface Standard=SDH; Virtual Concatenation=ENABLE; Represented Signal Structure: 7 times VC4
- Values: N = 64, X = 4; Modifier=10GEPL; Interface Standard=SDH; Virtual Concatenation=ENABLE; Represented Signal Structure: 64 times VC4
- Values: N = 1, X = c64; Modifier=10GEPL; Interface Standard=SDH; Virtual Concatenation=DISABLE; Represented Signal Structure: 1 times VC4-64C
- Values: N = 21, X = 1; Modifier=omitted; Interface Standard=SONET; Virtual Concatenation=not applicable; Represented Signal Structure: 21 times STS-1
- Values: N = 21, X = 1; Modifier=1GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 21 times STS-1
- Values: N = 7, X = 3; Modifier=1GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 7 times STS-3C
- Values: N = 192, X = 1; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 921 times STS-1
- Values: N = 64, X = 3; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=ENABLE; Represented Signal Structure: 64 times STS-3C
- Values: N = 1, X = 192; Modifier=10GEPL; Interface Standard=SONET; Virtual Concatenation=DISABLE; Represented Signal Structure: 1 times STS-192C

For *modifier* = ES64U, the value combination of “N” and “X” is done according to the following rules:

- *istd* = SONET
- X = 1 for STS-1
- N $\in$ {1, 2, ..., 192}

For *modifier* = ES64U, the parameter *inputsig* is visible for retrieval, but it cannot be modified once the VCG is established.

Default:

- modifier = omitted, Interface Standard=SDH, Default Value = 74
- modifier = 1GEPL, Interface Standard=SDH, Default Value = 74
- modifier = 10GEPL, Interface Standard=SDH, Default Value = 644
- modifier = omitted, Interface Standard=SONET, Default Value = 211
- modifier = 1GEPL, Interface Standard=SONET, Default Value = 211
- modifier = 10GEPL, Interface Standard=SONET, Default Value = 1921
- modifier = ES64U, Interface Standard=SONET, Default Value not applicable

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, i.e. VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception: For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.
For *modifier* = ES64U, this parameter will always be defined by the **`ENT-VCG`** command.

11. **`istd`** – Interface Standard
This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
For *modifier* = 1GEPL, 10GEPL, or ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: Alphanumeric value set.
Range:
    - SDH
    - SONET.

    Default: *modifier*: 1GEPL, 10GEPL, or ES64U: The value is derived from the circuit pack parameter *istd*. no *modifier*: As derived from parameter *istddflt* specified at configuration management, section **`RTRV-EQPT`**.
12. **`lcasmd`** – LCAS Mode
This parameter specifies the operation mode of the Link Capacity Adjustment Scheme for this VCG. This value is bidirectional.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: Alphanumeric value set.
Range:
    - DISABLE
    - ENABLE

    Default:
    - *modifier* = ES64U: DISABLE
    - other or no *modifier*: ENABLE

13. **`localLPID`** – Local Logical Port IDentity
    The *localLPID* provides a unique logical identification of a single port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like e.g. *networkTNAAddress*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkLPID", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    This parameter is supported by ports of *eportClass* UNI.
    The parameter *localLPID* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: numeric value set.
    Range: 0 to 4294967295.
    Default: 0.
14. **`localLPIDVCAT`** – Local Logical Port Identity on VCAT layer
    The *localLPIDVCAT* provides a unique logical identification of a single VCG port with an *eportClass* value used by the ONNS application and supporting this specific parameter. This identification is from the local NEs point-of-view.
    Depending on the *eportClass* value of the port, the uniqueness may be defined as relative to the whole node or relative to the value of any other ONNS related port parameter like, for example, *networkTNAAddrVCAT*.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *localLPIDVCAT* is used by ONNS for switching network connections on VCAT layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: numeric value set.
    Range: 0 to 4294967295.
    Default: 0.

15. **`macaddr`** – Port MAC Address
    This parameter specifies the MAC address of the WAN port.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
    If the parameter is not supported, is shall be omitted.
    Type: Alphanumeric value set.
    Range: 000000000000 to FFFFFFFFFFFF or UNAVAILABLE if the value can not be retrieved.
16. **`networkTNAType`** – Network TNA Type
    The *networkTNAType* identifies the type of the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAType", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *networkTNAType* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.
    Range: IPV4, IPV6, or NSAP.
    Default: IPV4.
17. **`networkTNATypeVCAT`** – Network TNA Type on VCAT level
    The *networkTNATypeVCAT* identifies the type of the Network TNA Address on VCAT level attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *networkTNATypeVCAT* is used by ONNS for switching network connections on VCAT layer.
    This parameter is supported by ports of eportClass UNI.
    This parameter is supported by ports of eportClass EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type: Alphanumeric value set.
    Range: IPV4, IPV6, or NSAP.
    Default: IPV4.

18. **`networkTNAAddress`** – Network TNA Address
    The *networkTNAAddress* identifies the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given NetworkTNAType. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, i.e. it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    This attribute replaces attribute "uniNetworkTNAAddress", as this attribute will not be exclusively valid any more for *eportClass* UNI only and thus a more generic name seems reasonable.
    The parameter *networkTNAAddress* is used by ONNS for switching network connections on SONET/SDH layer.
    This parameter is supported by ports of *eportClass* UNI.
    This parameter is supported by ports of *eportClass* EDGE.
    For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
    Type:
    networkTNAType= IPV4: Four dot separated numeric string (represents 4 Bytes)
    networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 Bytes)
    networkTNAType= NSAP: Hexadecimal number (represents 20 Bytes)
    Range:
    IPV4: Each numeric string: 0…255
    IPV6: Each hexadecimal number: 0000…FFFF
    NSAP: Hexadecimal number:
    0000000000000000000000000000000000000000 …
    FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
    Examples:
    IPV4: 129.12.3.178
    IPV6: AE12-0-0-12-EDFB-89AB-2254-8A2E
    NSAP: FFAABBCCDD213141515A6A7EB2D433221100AAFF
    Default:
    IPV4: 0.0.0.0
    IPV6: 0-0-0-0-0-0-0-0
    NSAP: 0000000000000000000000000000000000000000
19. **`networkTNAAddrVCAT`** – Network TNA Address on VCAT level
    The *networkTNAAddrVCAT* identifies the Network TNA Address attached to a port with an *eportClass* value used by the ONNS application and supporting this specific parameter.
    The content depends on the given *NetworkTNATypeVCAT*. For all ports with *eportClass* values not supporting this parameter, the parameter is not relevant, that means, it can be retrieved and changed, but will internally not be evaluated until *eportClass* is set to a value supporting it.
    The parameter *networkTNAAddressVCAT* is used by ONNS for switching network

connections on VCAT layer.
This parameter is supported by ports of *eportClass* UNI.
This parameter is supported by ports of *eportClass* EDGE.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type:
networkTNAType= IPV4: Four dot separated numeric string (represents 4 bytes)
networkTNAType= IPV6: Eight dash separated hexadecimal number (represents 16 bytes)
networkTNAType= NSAP: Hexadecimal number (represents 20 bytes)
Range:
IPV4: Each numeric string: 0…255
IPV6: Each hexadecimal number: 0000…FFFF
NSAP: Hexadecimal number:
0000000000000000000000000000000000000000 …
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
Default:
IPV4: 0.0.0.0
IPV6: 0-0-0-0-0-0-0-0
NSAP: 0000000000000000000000000000000000000000

20. **plcrthr** – Threshold for Partial Loss of Capacity (Receive)
The parameter *plcrthr* specifies the acceptable limit of valid VCG members in receive direction (refer to the *outmbs* parameter of all related VCGTRIB entities) for which the PLCR alarm will not yet be raised. The PLCR alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plcrthr* parameter – 1)] (if *plcrthr* ≥ 2). The PLCR alarm shall never be raised if *plcrthr* is set to 1.
The PLCR alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plcrthr* parameter (or if the number of those VCG members is 0 – in this case the TLCR alarm will be raised instead). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plcrthr* is not supported, then the parameter is omitted.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 1, 2, 3, …, 192
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192

Modifier	Input signal structure	Default value
ES64U	n × 1	n

21. **`plctthr`** – Threshold for Partial Loss of Capacity (Transmit)
The parameter *plctthr* specifies the acceptable limit of valid VCG members in transmit direction (refer to the *inmbs* parameter of all related VCGTRIB entities) for which the PLCT alarm will not yet be raised. The PLCT alarm will be raised if the actual number of those VCG members is between [1 … (value of the *plctthr* parameter – 1)] (if *plctthr* ≥ 2). The PLCR alarm shall never be raised if *plctthr* is set to 1.
The PLCT alarm will be cleared if the actual number of those VCG members is equal to or more than the value of the *plctthr* parameter (or if the number of those VCG members is 0 – in this case the TLCT alarm will be raised instead).
This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *plctthr* is not supported, then the parameter is omitted.
For *modifier* = ES64U, the parameter is visible for retrieval, but it cannot be modified.
Type: numeric value set.
Range: 1, 2, 3, …, 192
Default:

Modifier	Input signal structure	Default value
1GEPL	74	7
1GEPL	211	21
10GEPL	644, 1c64	64
10GEPL	1921, 1192	192
ES64U	n × 1	n

22. **`stdis`** – Spanning Tree Protocol Disable
This parameter allows to enable or disable the Spanning Tree Protocol for the given port. If the input parameter modifier is omitted, then this parameter is supported but applicable only if the port is a network port (cnprole = NETWORK). For customer ports (cnprole = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stdis* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range: DISABLE – STP is disabled, that means, the port is not used to build the spanning tree. ENABLE – STP is enabled, that means, the port is used to build the spanning tree.
Default: ENABLE.
As VCGs are by default network ports the default value is to enable STP.

23. **`stmd`** – Spanning Tree Protocol Mode
This parameter returns whether the normal STP according to IEEE802.1D or the rapid spanning tree protocol rSTP according to IEEE802.1w is used on the given port.
This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER), this parameter has no effect. However, changes are stored in the database.
If the input parameter *modifier* = 1GE, then this parameter is applicable meaningful only if the port is a network port (*cnprole* = NETWORK). For customer ports (cnprole = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stmd* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range:
   - 802.1D, for the normal ST
   - 802.1W, for rSTP
   - UNAVAILABLE if the value cannot be retrieved.

   Default: There is no default value.

24. **`stpcost`** – Spanning Tree Path Cost
The parameter *stpcost* specifies the STP-related cost of the path egressing the given port.
If the input parameter *modifier* is omitted, then this parameter is supported but applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
If the parameter *stpcost* is not supported, then the parameter is omitted.
Type: Integer.
Range: 1 to 200000000.
Default: 20000.
The default value depends on the provisioned VCG size. As currently only a fix size of 7 VC4s or 21 STS1s corresponding to approximately 1 Gbit is supported, the default value is as shown above.

25. **`stpid`** – Spanning Tree Port Identifier
    The port identifier is a 16 bit parameter consisting of the port priority (high-order bits) and the port number (low-order bits). The port number is assigned by the system to a port and is not provisionable.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stpid* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: 0 to 65535 and UNAVAILABLE if the value can not be retrieved.
26. **`stpop`** – Spanning Tree Protocol Operational
    This parameter indicates whether the Spanning Tree Protocol is operational or not.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stpop* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range: YES, operational; NO, not operational, UNAVAILABLE if the value can not be retrieved.
27. **`stpp`** – Spanning Tree Port Priority
    The parameter *stpp* specifies the priority of the given port for the Spanning Tree Protocol.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stpp* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 0 to 255.
    Default: 128.
28. **`stprole`** – Spanning Tree Protocol Role
    The parameter *stprole* shall report the role of the given port in rSTP.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK) and operates in rSTP mode (*stmd* = 802.1W). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However,

changes are stored in the database.
This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *stprole* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.

Range:

- ROOT, root
- DESG, designated
- ALT, alternative
- BACKUP, backup
- UNAVAILABLE if the value can not be retrieved.

29. **`tcamd`** – TCA Mode
This parameter shows the modus of the PM thresholding.
Two thresholding methods are applicable: TR-only, (also known as the implicit reset method). TR-RTR, (also known as the explicit reset method).
This parameter is not supported if the input parameter *modifier* = ES64U. If the parameter *tcamd* is not supported, then the parameter is omitted.
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: TRRTR in SDH mode; TR in SONET mode.

30. **`unequiposig`** – Unequipped Tributary Output Signal Rate List
The parameter *unequiposig* specifies the tributary structure and the capacity of the VCG port in output direction. As long as there is no cross-connection applied the specified tributaries are occupied by an unequipped signal. The parameter will implicitly give the maximum capacity of the VCG port, too.
If the input parameter *modifier* is omitted, then the tributary structure depends on interface standard specified for this VCG by *istd* . For the definition of *istd* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078).
If the input parameter *modifier* = 1GEPL or *modifier* = 10GEPL, then the range of possible tributary structures depends on interface standard specified for this VCG by *istd* and on the virtual concatenation mode *vcat*. For definitions of *istd* and *vcat* refer to the “Output parameters” section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). The signal will be immediately terminated on the unit. So no unequipped signal is visible to the outside world.
If the input parameter: *modifier* = ES64U, then the parameter *unequiposig* is set automatically according to the value of the *inputsig* parameter.
For *modifier* = ES64U, the parameter *unequiposig* is only retrievable and cannot be changed because it follows always the value of the *inputsig* parameter.

Type: numeric string of the form ***NX***

- ***N*** specifies the count of container type ***X***
- ***X*** specifies the container type (or envelope).

Range:

Values	Modifier	Interface standard	Virtual concatenation mode	Represented signal structure
N = 7, ***X*** = 4	omitted	SDH	not applicable	7 × VC4
N = 7, ***X*** = 4	1GEPL	SDH	ENABLE	7 × VC4
N = 64, ***X*** = 4	10GEPL	SDH	ENABLE	64 × VC4
N = 1, ***X*** = c64	10GEPL	SDH	DISABLE	1 × VC4-64C
N = 21, ***X*** = 1	omitted	SONET	not applicable	21 × STS-1
N = 21, ***X*** = 1	1GEPL	SONET	ENABLE	21 × STS-1
N = 7, ***X*** = 3	1GEPL	SONET	ENABLE	7 × STS-3C
N = 192, ***X*** = 1	10GEPL	SONET	ENABLE	192 × STS-1
N = 64, ***X*** = 3	10GEPL	SONET	ENABLE	64 × STS-3C
N = 1, ***X*** = 192	10GEPL	SONET	DISABLE	1 × STS-192C

For *modifier* = ES64U, the same value set is applicable as for the parameter *inputsig*.

Default:

Modifier	Interface Standard	Default Value
omitted	SDH	74
1GEPL	SDH	74
10GEPL	SDH	644
omitted	SONET	211
1GEPL	SONET	211
10GEPL	SONET	1921
ES64U	SONET	not applicable

Default in case of System Default *portClass* EDGE: In case of the *portclassdefault* EDGE, that means, VCGs are created with *eportClass* EDGE, the default values for *unequiposig* is as in table above with the following exception:

For *modifier* = GE10PL and *Interface Standard* = SONET, the default value is 643.

31. **`vcat`** – Virtual Concatenation
    The parameter *vcat* specifies whether the Ethernet signal is mapped to a number of virtually concatenated tributaries (value = ENABLE) or into a contiguously concatenated tributaries (value = DISABLE). In the latter case, LCAS related parameters become insignificant (values set to default, values cannot be changed). This parameter is not supported if the input parameter *modifier* is omitted. If the parameter *vcat* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - For *modifier* = ES64U: ENABLE.
    - Otherwise: ENABLE, DISABLE.

    For *modifier* = ES64U, the parameter *vcat* is visible for retrieval, but cannot be modified once the VCG is established.
    Default: Enable.
32. **`vmode`** – VCG Monitoring Mode
    the parameter *vmode* specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range:
    - MON, for monitored
    - NMON (default), for not monitored.
33. **`vpm`** – VCG Performance Monitoring
    The parameter *vpm* specifies whether the performance monitoring is enabled or disabled for this port.
    This parameter is not supported if the input parameter *modifier* = . If the parameter *vpm* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - DISABLE (default)
    - ENABLE.
34. **`vrtswid`** – Virtual Switch Identifier
    This parameter identifies the instance of the virtual switch.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *vrtswid* is not supported, then the parameter is omitted.
    Type: Integer.
    Range: 1 – Currently this is the only allowed value

35. **`vss`** – Spanning Tree Status
    This parameter returns the status of the spanning tree protocol for the given port.
    This parameter is applicable only if the port is a network port (*cnprole* = NETWORK
). For customer ports (*cnprole* = CUSTOMER), this parameter has no effect. However, changes are stored in the database.
    If the input parameter *modifier* = 1GE, then this parameter is applicable only if the port is a network port (*cnprole* = NETWORK). For customer ports (*cnprole* = CUSTOMER) this parameter has no effect. However, changes are stored in the database.
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
    If the parameter *vss* is not supported, then the parameter is omitted.
    Type: Alphanumeric value set.
    Range:
    - FWG, Forwarding
    - LRG, Learning
    - LIS, Listening (IEEE8102.D/STP only)
    - BLK, Blocking (IEEE8102.D/STP only)
    - DISF, Disabled due to a failure (IEEE8102.D/STP only)
    - DISA, Disabled via provisioning
    - BRK, Broken (IEEE8102.w/rSTP only)
    - DCD, Discarding (IEEE8102.w/rSTP only)
    - UNAVAILABLE if the value can not be retrieved.
36. **`vvidls`** – Valid VLAN Identifier List (static)
    This parameter specifies a list of all static VLAN identifiers allowed for this port. The list is ampersand grouped (& and &&). The list is packed by the NE and the management systems as good as possible by using '&' and '&&' . The vlan identifier of an incoming frame must be a member of this static VLAN Id list or the dynamic VLAN Id list *vvidlsd*. Frames tagged by a vlan identifier not member of these lists will be discarded. The port VLAN identifier (refer to parameter dftvlan of RTRV-EPORT Output Parameters) is not part of this list. This parameter is also used in transparent tagging mode to specify the list of valid customer tags (CIDs) for network ports (*cnprole* = NETWORK).
    This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U. If the parameter *vvidls* is not supported, then the parameter is omitted.
    Type: Alphanumeric string and the character '&'.

Range:

- Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
- In case the list has no members (empty list) –1.

Default: –1.

37. **`vvidlsd`** – Valid VLAN Identifier List (dynamic)
This parameter gives the list of all dynamic VLAN identifiers allowed for this port as gathered autonomously via GVRP. The list is ampersand grouped (& and &&). The vlan identifier of an incoming frame must be a member of this dynamic VLAN Id list or the static VLAN Id list vvidls. Frames tagged by a vlan identifier not member of these lists will be discarded. This parameter is also used in transparent tagging mode to report the list of valid customer tags (CIDs) for network ports (*cnprole* = NETWORK) collected by GVRP.
This parameter is not supported if the input parameter *modifier* = 1GEPL, 10GEPL, or ES64U.
Ths parameter is reported only if
    - the virtual switch entity exists (i.e. the VLAN tagging mode *vlantagmd* of the slot the addressed EPort belongs to is set to 802.1TAG) AND
    - GVRP is enabled on the virtual switch (*gvrpmode* = ENABLED)
    - the port is a NETWORK port (*cnprole* = NETWORK).

    Type: Alphanumeric string and the character "&".
    Range:
    - Value of each single member (valid VLAN identifier) in the list 1, 2, …, 4092, 4093
    - In case the list has no members (empty list) –1.

    If the value can not be retrieved: UNAVAILABLE
    Default: There is no default value.
38. **`wtr`** – Wait To Restore Time
The parameter *wtr* specifies the wait to restore time for the LCAS protection. If a tributary is configured to take part on transmission for the VCG and it recovers from a faulty state then this tributary is taken in again to transmission by LCAS after the wait-to-restore time. This time is configured on group level, i.e. all tributaries have the same value for the wait-to-restore time. This parameter only affects NE behavior if *lcasmd* = ENABLED.
For *modifier* = ES64U, the parameter is visible with default values for retrieval, but cannot be modified.
Type: Alphanumeric value set.
Range: 0, 1, 2, …, 59, 60 [minutes].
Default: 0.

Detailed behavior description

The **RTRV-VCG** does not generate a **REPT DBCHG** message. The command retrieves all VCG parameters as described in section Output parameters.

The values of the following parameters are directly requested form the hardware:

- mfiv
- macaddr
- stmd
- stpop
- stpid
- stprole
- vss
- vvidlsd.

Example input/output

Example of a VCG retrieval on a GE1 pack working in switched mode:

```
RTRV-VCG:UNITE-NE:1-1-#-#-2-v3:123456;

   UNITE-NE 08-03-28 09:05:00
M 123456 COMPLD
   ”1-1-#-#-02-
      v3:cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT0,eportCl
   ”1-1-#-#-02-v3-1:125,MFIV”
   .
   . for all 7 VCG Tributaries belonging to the VCG
   .
   ”1-1-#-#-02-v3-19:250,MFIV”
;
```

Example of a retrieval of all VCGs on a GE1 pack:

```
RTRV-VCG:UNITE-NE:1-1-#-#-2-vall:123456;

   UNITE-NE 08-03-28 09:05:00
M 123456 COMPLD
   ”1-1-#-#-02-
      v1:cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT0,eportCl
”1-1-#-#-02-
      v2:cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT0,eportCl
”1-1-#-#-02-
      v3:cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT0,eportCl
”1-1-#-#-02-
      v4:cnprole=NETWORK,dftpri=LOW,dftvlan=UNAVAILABLE,enetpn=DEFAULT,enettca=DEFAULT0,eportCl
;
```

Example of a VCG retrieval related to an UNI port on a GE10PL1 pack:

```
RTRV-VCG-1GE:UNITE-NE:1-1-#-#-39-v1:123456;

   UNITE-NE 08-03-28 09:05:00
M 123456 COMPLD
   ”1-1-#-#-39-v1:enetpn=DEFAULT,enettca=DEFAULT0,eportClass=UNI,\
    eportControl=NORMAL,htmr=0,inputsig=211,istd=SONET,lcasmd=ENABLE,\
    plcrthr=21,plctthr=21,tcamd=TR,unequiposig=211,vcat=ENABLE,vmode=NMON,\
    vpm=DISABLE,wtr=0,networkTNAType=IPV4,networkTNAAddress=0.0.0.0,\
    localLPID=0,clientNodeID=12.1.31.11,networkTNATypeVCAT=IPV4,\
    networkTNAAddrVCAT=111.121.23.6,localLPIDVCAT=12”
   ”1-1-#-#-39-v1-1:2000,MFIV”
   .
   . for all 21 VCG Tributaries belonging to the VCG
   .
   ”1-1-#-#-39-v1-21:8000,MFIV”
;
```

Example of a VCG retrieval related to a 1 GE interface on a GE10PL1 pack (virtual concatenation mode enabled):

```
RTRV-VCG-1GE:UNITE-NE:1-1-#-#-39-v1:123456;

   UNITE-NE 08-03-28 09:05:00
M 123456 COMPLD
   ”1-1-#-#-39-v1:enetpn=DEFAULT,enettca=DEFAULT0,eportClass=TRADITIONAL,\
    eportControl=NORMAL,htmr=0,inputsig=211,istd=SONET,lcasmd=ENABLE,\
    plcrthr=21,plctthr=21,tcamd=TR,unequiposig=211,vcat=ENABLE,vmode=NMON,
    vpm=DISABLE,wtr=0,networkTNAType=IPV4,networkTNAAddress=0.0.0.0,\
    clientNodeID=0.0.0.0,networkTNATypeVCAT=IPV4,localLPID=0,\
    networkTNAAddrVCAT=0.0.0.0,localLPIDVCAT=0”
   ”1-1-#-#-39-v1-1:2000,MFIV”
   .
   . for all 21 VCG Tributaries belonging to the VCG
   .
   ”1-1-#-#-39-v1-21:8000,MFIV”
;
```

Example of a VCG retrieval related to a 10 GE interface on a GE10PL1 pack (virtual concatenation mode disabled):

```
RTRV-VCG-10GEPL:LT-UNITE:1-1-#-#-39-v9:123456;

   LT-UNITE 01-03-20 09:05:00
M 123456 COMPLD
```

```
   ″1-1-#-#-39-v9:enetpn=DEFAULT,enettca=DEFAULT0,eportClass=TRADITIONAL,\
    eportControl=NORMAL,htmr=0,inputsig=1c64,istd=SDH,lcasmd=ENABLE,\
    plcrthr=64,plctthr=64,tcamd=TRRTR,unequiposig=1c64,vcat=DISABLE,\
    vmode=NMON,vpm=DISABLE,wtr=0″
;
```

Example of a VCG retrieval related to an ES64U pack:

```
RTRV-VCG-ES64U:UNITE-NE:1-1-#-#-35-v1:123456;

   UNITE-NE 08-03-28 09:05:00
M 123456 COMPLD
   ″1-1-#-#-35-v1: enetpn=DEFAULT,eportClass=TRADITIONAL,\
    eportControl=NORMAL,htmr=0,inputsig=211,istd=SONET,lcasmd=DISABLE,\
    plcrthr=21,plctthr=21,unequiposig=211,clientNodeID=0.0.0.0,\
    vcat=ENABLE,vmode=NMON,wtr=0,networkTNAType=IPV4,\
    networkTNAAddress=0.0.0.0,localLPID=0,networkTNATypeVCAT=IPV4,\
    networkTNAAddrVCAT=0.0.0.0,localLPIDVCAT=0″
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-VCG** command, too.

Validation rule: Command only allowed for VCG on GE*n* or ES64U circuit packs

If the command is not applied to a VCG on a GE*n* or ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: *modifier* mandatory for the GE10PL1 and ES64U circuit pack

The system shall verify that the modifier parameter is provided by the command if the related slot has been successfully provisioned with a GE10PL1 or ES64U circuit pack.

Error code:

`IDNC – Input, Data Not Consistent, modifier is required for this circuit pack type`

Validation rule: *modifier* consistent with given port-rate

The system shall verify that a given modifier parameter is consistent with the rate of the port (given by the input aid parameter). Refer to the Range definition of the input parameter modifier of the **RTRV-VCG** command for applicable combinations.

Error code:

`IDNC – Input, Data Not Consistent, modifier is inconsistent with existing port rate`

Related TL1 messages

- "DLT-VCG – Enter virtual concatenation group" (p. 3-277)
- "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
- "ENT-VCG – Enter virtual concatenation group" (p. 3-1983)

RTRV-VCG-TRANS: Retrieve virtual concatenation group transparent

The **RTRV-VCG-TRANS** command retrieves the attributes of a virtual concatenation group, used for fully transparent applications.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes.

Input format

Command syntax: **RTRV-VCG-TRANS***:tid:aid:ctag[:::spec_block];*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the virtual concatenation group (VCG) to be operated on.
   Type: AID.

   Range: VCGAID as defined in AID structure. The values

   - *vall* for Simple_VCGAID and
   - *usall* for Simple_UnivSlotAID

   are accepted.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameter

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values usall or vall at the appropriate positions of the input AID. For input AIDs not using usall or vall this parameter is not considered. This implies that with such input AIDs the dynamic parameters are always returned.
   The following output parameters of a vcgport port are affected: *mfiv*, *rplt*.
   Type: Alphanumeric value set.
   Range:
   - YES: Report dynamic parameters in the reply as well
   - NO (default): Do not report dynamic parameters in the reply.

Output format

If the NE fully complies with the RTRV-VCG request, the following normal completion response is returned:

```
sid date time
M  ctag COMPLD
   ”aid::spec_block”
      <1 or more of the line above>
   ”tribaid:mfiv,MFIV”
    <no such line, if the virtual concatenation mode is disabled, else
     no such line, if no dynamic data are retrieved, else
     one or more lines, one for each tribaid belonging to aid>
       ...
       ...
   ”aid::spec_block”
      <1 or more of the line above>
   ”tribaid:mfiv,MFIV”
    <no such line, if the virtual concatenation mode is disabled, else
     no such line, if no dynamic data are retrieved, else
     one or more lines, one for each tribaid belonging to aid>
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the VCG to be operated on.
   Type: AID.
   Range: VCGAID as defined in AID structure with the exception of the value *vall*.
2. **`tribaid`** – Tributary Access Identifier
   The tributary access identifier specifies a tributary belonging to the virtual concatenation group (VCG) given by the aid above.
   Type: AID.
   Range: VCGNumAID as defined in AID structure with the exception of the value *all*.
3. **`mfiv`** – Multiframe Indicator Value
   This parameter reports the differential delay between the tributary addressed by *tribaid* (see above) and the fastest tributary of the VCG. The fastest tributary has the value zero. For each retrieve of a VCG there is a report of *mfiv* for all tributaries belonging to the VCG.
   Type: Alphanumeric value set.

   Range:

   - 0, 125, 250, 375, …, 511750, 511875 [microseconds]
   - NA
   - UNAVAILABLE

   In fact the reported range for delay is 0 … 255875 microseconds. Other values can only be reported as raw values without a delay calculation between the tributaries. UNAVAILABLE is reported when the accepted MFI contents do not meet the basic acceptance criteria (multiframe alignment not completed), or intra-NE communication with the reporting entity is disrupted. NA (Not Applicable) is reported under defect conditions detected by the Sn[i] termination function or the VC-n is not part of an S-n-Xv group.

Spec block parameters

1. **`camd`** – Consequent Action Mode
   This parameter specifies the mode of the consequent action to be executed in egress direction (to the synchronous port) on detected signal fail conditions (dPLM, AI_TSF or detected generic AIS) on the ODU layer
   Type: Alphanumeric value set.
   Range: GENERICAIS (default), MSAIS, LASEROFF
2. **`enetpn`** – Ethernet Alarm Severity Assignment Profile Name
   This parameter points to an ASAP profile name of type ENET for the designated AID.
   Some VCG related alarms have to be referenced via the ASAP profile named ENET.
   Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
   Range: 1 … 24 characters length.
   Default: DEFAULT.
3. **`fevpm`** – Far End VCG Performance Monitoring
   The parameter fevpm specifies whether the far end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer. This parameter enables/disable therefore far end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
4. **`inputsig`** – Tributary Input Signal Rate List
   This parameter reports the tributary structure of the VCG port in input direction. This will implicitly give the maximum capacity of the VCG port, too. The tributary structure is independent from the interface standard istd specified for this VCG.

   Type: numeric string of the form NX

   - N specifies the count of container type X
   - X specifies the container type (or envelope).

   Range:

   - N=17,
   - X = 4 for VC-4 / STS3.

   Default: 174.
5. **`istd`** – Interface Standard
   This parameter specifies the interface standard of this VCG. This value is applied for both input and output direction.
   Type: Alphanumeric value set.

   Range:

   - SDH
   - SONET.

   Default: As derived from parameter *istddflt* specified at configuration management, section System RTRV-EQPT.

6. **`nevpm`** – Near End VCG (ODUkP) Performance Monitoring
   This parameter specifies whether the near end performance monitoring is enabled or disabled for this port. For the fully transparent application performance monitoring is actually done on the ODUkP layer and on the VCG level. This parameter enables/disable therefore near end PM on the ODUkP layer.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE
7. **`pmvcgfeses`** – Performance Monitoring VCG (ODUkP) Far-End SES Threshold
   This parameter describes the ODUkP Far-End SES threshold value. It indicates the number of errored blocks per second to declare a second as severely errored. The default value is defined in G.8201 for OTN. If the interface standard is changed the current default value is not changed.
   Type: numeric value set.
   Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
   Default: ODU1P: 3064
   G.8201 defines block based PM on the ODUkP layer.
8. **`pmvcgneses`** – Performance Monitoring VCG (ODUkP) Near-End SES Threshold
   This parameter describes the VCG (actually ODUkP) Near-End SES threshold value. It indicates the number of errored blocks (SDH) or code violations (SONET) per second to declare a second as severly errored. The default value depends on the interface standard and may differ between SDH and SONET. If the interface standard is changed the current default value is not adapted automatically.
   Type: numeric value set.
   Range (the range depends on the ODUkP rate): ODU1P: 1 – 20420
   Default: ODU1P: 3064
   G.8201 defines block based PM on the ODUkP layer.
9. **`rplt`** – Reported Payload Type
   The parameter rplt shall report the received payload type for the constant bit rate mapping on the ODU layer.
   Type: Alphanumeric value set.
   Range:
   - 00, 01, …, FE, FF i.e. two hexadecimal digits
   - UNAVAILABLE
10. **`tcamd`** – TCA mode
    This parameter shows the modus of the PM thresholding.
    Two thresholding methods are applicable: TR-only, (also known as the implicit reset method). TR-RTR, (also known as the explicit reset method).
    Type: Alphanumeric value set.
    Range: TR, TRRTR.
    Default: TRRTR in SDH mode; TR in SONET mode.

11. **transpn** – Transparent Alarm Severity Assignment Profile Name
    This parameter points to a ASAP profile name of type TRANSPARENCY for the designated AID.
    Type: Alphanumeric string. A hyphen ("-") is allowed in the name additionally.
    Range: 1 … 24 characters length
    Default: DEFAULT.
12. **transtca** – Transparent TCA Profile Name
    The parameter transtca specifies the name of the threshold crossing alert (TCA) profile used for the VCG..
    Type: Alphanumeric string.
    Range: 1 … 24 characters length
    Default: DEFAULT0
13. **unequiposig** – Unequipped Tributary Output Signal Rate List
    This parameter specifies the tributary structure and the capacity of the VCG port in output direction. As long as there is no cross-connection applied the specified tributaries are occupied by an unequipped signal. The parameter will implicitly give the maximum capacity of the VCG port, too. The tributary structure is independent of the interface standard specified for this VCG.
    The signal will be immediately terminated on the unit. So no unequipped signal is visible to the outside world.

    Type: numeric string of the form NX.

    - N specifies the count of container type X
    - X specifies the container type (or envelope)

    Range:

    - N = 17
    - X = 4 for VC-4 / STS-3C

    Default: 174
14. **vmode** – VCG Monitoring Mode
    This parameter wtr specifies the monitoring mode for VCG. Alarms for this port will be reported if in mode MON otherwise not. There is no automatic mode.
    Type: Alphanumeric value set.
    Range: MON: for monitored; NMON (default): for not monitored.

Detailed behavior description

The RTRV-VCG-TRANS does not generate a **REPT DBCHG** message. The command retrieves all VCG parameters as described in section Output parameters.

The values of the following parameters are directly requested form the hardware:

- *mfiv*
- *rplt*

Thus the value ″UNAVAILABLE″ is reported, if the CP HW is not available in the system.

Example input/output

Example 1:

```
RTRV-VCG-TRANS:LT-UNITE:1-1-#-#-2-v3:123456;

   LT-UNITE 01-03-20 09:05:00
M 123456 COMPLD
   ”1-1-#-#-02-
      v3:camd=GENERICAIS,enetpn=DEFAULT,fevpm=DISABLE,inputsig=174,istd=SDH,nevpm=DISABLE,pmvcg
   ”1-1-#-#-02-v3-1:125,MFIV”
   .
   . for all 17 VCG Tributaries belonging to the VCG
   .
   ”1-1-#-#-02-v3-17:250,MFIV”
;
```

Example 2:

```
RTRV-VCG-TRANS:LT-UNITE:1-1-#-#-2-vall:123456;

   LT-UNITE 01-03-20 09:05:00
M 123456 COMPLD
   ”1-1-#-#-02-
      v1:camd=GENERICAIS,enetpn=DEFAULT,fevpm=DISABLE,inputsig=174,istd=SDH,nevpm=DISABLE,pmvcg
   ”1-1-#-#-02-
      v2:camd=GENERICAIS,enetpn=DEFAULT,fevpm=DISABLE,inputsig=174,istd=SDH,nevpm=DISABLE,pmvcg
   ”1-1-#-#-02-
      v3:camd=GENERICAIS,enetpn=DEFAULT,fevpm=DISABLE,inputsig=174,istd=SDH,nevpm=DISABLE,pmvcg
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-VCG-TRANS** command, too.

Validation rule: Command only allowed for VCG on Transparent Packs

If the command is not applied to a VCG on a transparent pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Related TL1 messages

“ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application” (p. 3-1405)

RTRV-VCGTRIB: Retrieve virtual concatenation group tributary

The **RTRV-VCGTRIB** command retrieves the attributes of a VCG tributary used for 1 and 10 GbE and ES64U or fully transparent applications.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-VCGTRIB**-*modifier*:*tid*:*aid*:*ctag*[:::*spec_block*];

Input parameters

Command parameters

1. **modifier**
   This parameter specifies the type of the tributary. The purpose is to distinguish between SDH/SONET tributaries and between the application of the VCG the tributary belongs to. Current applications are 1 and 10 GbE and ES64Uand fully transparency.
   Type: Alphanumeric value set.
   Range:
   - ALL
   - STS
   - VC.
   - STST
   - VCT.

   To distinguish the fully transparent application from a GE application, a letter ″T″ is appended. There is no distinction between VCGTRIB entities of the GE1 pack from the GE10PL1 or ES64U pack needed. In particular, the transmission rate in case the parameter *modifier* = STS needs to be derived from the parameters inputsig and unequiposig of the related VCG entity.
2. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
3. **aid** – Access Identifier
   This parameter specifies VCG tributary to be operated on.
   Type: AID.

Range: VCGNumAID as defined in AID structure. The values

- *all* for Simple_VCGNumAID
- *vall* for Simple_VCGAID
- *usall* for Simple_UnivSlotAID
- *all* for Simple_UnivSlotAID
- *all* for Simple_ShelfAID and
- *all* for BayAID

is accepted.

4. **ctag** – Correlation Tag
This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
Type: Alphanumeric string
Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.

Spec block parameter

dyn – Retrieve Dynamic Parameters

This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values all, usall or vall at the appropriate positions of the input AID. For input AIDs not using all, usall or vall this parameter is not allowed.

The following output parameters of a vcgtrib are affected:

- aseqno
- tseqno
- ptrc
- ptrcst
- rsiglb
- trailst.

Type: Alphanumeric value set.

Range: YES: Report dynamic parameters in the reply as well; NO (default): Do not report dynamic parameters in the reply.

Output format

If the NE fully complies with the RTRV-VCGTRIB request, the following normal completion response is returned:

```
sid date time
M  ctag COMPLD
   ″aid::spec_block set1″
```

```
    <1 or more of the line above,
     or no lines, depending on dynamic data retrieval mode>
   ″aid,aidtype::spec_block set2″
    <1 or more of the line above>
       ...
       ...
   ″aid::spec_block set1″
    <1 or more of the line above,
     or no lines, depending on dynamic data retrieval mode>
   ″aid,aidtype::spec_block set2″
    <1 or more of the line above>
;
```

The output parameters *sid*, *date*, *time* and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the VCG tributary whose attributes are reported.
   Type: AID.
   Range: VCGNumAID as defined in AID structure with exception of value all.
2. **aidtype** – Access Identifier Type
   This parameter specifies the type of the tributary.
   Type: Alphanumeric value set.
   Range: STS or VC or STST or VCT.

To distinguish the fully transparent application from a GE application, a letter ″T″ is appended. There is no distinction between VCGTRIB entities of the GE1 pack from the GE10PL1 or ES64U pack needed. In particular, the transmission rate in case the parameter *modifier* = STS needs to be derived from the parameters inputsig and unequiposig of the related VCG entity.

Spec block parameters Set 1

1. **aseqno** – Accepted Sequence Number
   This parameter reports the accepted sequence number of the tributary. In case of a disrupted intra-NE communication with the reporting entity “not available” is reported.
   In case that the received sequence no. does not meet the basic acceptance criteria or under defect conditions a value is reported, which is outside of the designated range (see parameter *tseqno* for the designated range).
   If the tributary to be reported is no configured member of the VCG “not applicable” is reported.
   Type: Alphanumeric value set.

Range:

- 0, 1, 2, ..., 254, 255
- NA, for not applicable
- UNAVAILABLE, for not available/unavailable.

  Although the normal accepted sequence no. is limited to the designated range (see parameter *tseqno* for the designated range), the NE is not prevented to report a higher number (accompanied by a proper alarm).

2. **tseqno** – Transmitted Sequence Number

   This parameter reports the transmitted sequence number of the tributary. In case of a disrupted intra-NE communication with the reporting unit 'not available' is reported. If the tributary to be reported is no configured member of the VCG 'not applicable' is reported.

   Type: Alphanumeric value set.

   Range:

   - For modifier STS: 0, 1, 2, ..., 190, 191
   - For modifier VC: 0, 1, 2, ..., 62, 63
   - For modifier STST or VCT: 0, 1, 2, ..., 15, 16
   - For modifier STS or VC: NA for not applicable
   - For all modifiers: UNAVAILABLE for not available/unavailable.

   The following table explains the possible value ranges for the various pack types:

Pack type	*modifier*	*tseqno* range	Remarks
GE1	STS	0, [1, ..., 20]	depends also on configured and actual membership
	VC	0, [1, ..., 6]	
GE10PL1	STS	0, [1, ..., 20]	if VCGTRIB entities belong to 1 GE interface
	VC	0, [1, ..., 6]	depends also on configured and actual membership
	STS	0, [1, ..., 191]	if VCGTRIB entities belong to 10 GE interface
	VC	0, [1, ..., 63]	depends also on configured and actual membership
OPT2G5	STST	0, 1, ..., 16	always all 17 are needed and no LCAS is supported
	VCT		
<all>	<all>	NA	if not applicable
		UNAVAILABLE	if not available/unavailable

Values for the parameters are derived from the VCGTRIB number as follows:

modifier	Tributary rate	*tseqno* calculation
STS	STS-1	VCGTRIB No. - 1

modifier	**Tributary rate**	***tseqno* calculation**
STS	STS-3c	(VCGTRIB No. - 1) / 3 for VCGTRIB in {1, 4, ..., 190} otherwise NA
VC	VC-4	(VCGTRIB No. - 1) / 3 for VCGTRIB in {1, 4, ..., 190} otherwise NA
STST	STS-3c	VCGTRIB No. - 1
VCT	VC-4	

Spec block parameters Set 2

Mode STS | VC

Some of the spec block parameters set 2 depend on the *modifier* (for RTRV) or *aidtype* (for ED, ED Not. and ENT Not.). Those parameters are only applicable for the given mode, i.e in the other mode they is neither reported nor modifiable.

1. **`brstintvl`**- Bursty Interval
   This parameter specifies the measurement period for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstintvl* has no affect to NE behavior. The parameter *brstintvl* is only valid for modifiers different from STST and VCT, .e. is omitted in the command response for STST and VCT.
   Type: Integer.
   Range: 2, 3, ..., 10 [seconds].
   Default: 7 [seconds].
2. **`brstthr`** – Bursty Error Threshold
   This parameter specifies the degradation threshold for degraded defect if *sdsfmode* is set to BURST. If *sdsfmode* is not set to BURST then *brstthr* has no affect to NE behavior. The parameter brstthr is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
   Type: Integer.
   Range: 5, 10, 15, ... 50 [%].
   Default: 30 [%].
3. **`crslpbkstat`** – Cross-connect Loopback Status
   This parameter reports the configuration of the cross-connection loopback of this tributary. For changing the loopback operation refer to “OPR-LPBK: Operate cross-connection loopback” (p. 3-2035).
   Type: Alphanumeric value set.
   Range: YES, for enabled loopback; NO, for disabled loopback.
4. **`degthr`** – Signal Degrade Threshold
   This parameter specifies the exponent to base 10 for building the degraded threshold level. The degraded threshold level is used for building the degraded defect if sdsfmode is set to POISSON. If *sdsfmode* is not set to POISSON then *degthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the

Output parameters section. The parameter degthr is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
Type: Integer.
Range: -5, -6, -7, -8, -9 i.e. range is 10exp-9 to 10exp-5.
Default: -6

5. **`dexcthr`** – Excessive Degrade Threshold
   This parameter specifies the exponent to base 10 for building the excessive threshold level. The excessive threshold level is used for building the excessive bit error defect if *sdsfmode* is set to POISSON. If *sdsfmode* is not set to POISSON then *dexcthr* has no affect to NE behavior. For the definition of *sdsfmode* refer to the Output parameters section. The parameter *dexcthr* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
   Type: Integer.
   Range: -3, -4, -5 i.e. range is 10exp-5 to 10exp-3.
   Default: -3.
6. **`exphptrc`** – Expected HP trace identifier
   This parameter specifies the Expected Higher Order Path Trace Identifier (J1). The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is only useful if ptrcrfmt is set to 16.
   For VCG tributaries located on ES64U packs this parameter is visible with default values for retrieval, but cannot be modified.
   Type: Hexadecimal number without a preceding “H”.
   Range: length, 32 hexadecimal digits; value, specific string acc. ITU/ETSI mode 1.
   Default: 89000000000000000000000000000000.
7. **`feppm`** – Far End Path PM Enable
   This parameter specifies whether the performance monitoring for the far end is enabled or disabled for this tributary. The parameter *feppm* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
   For VCG tributaries located on ES64U packs this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
   Type: Alphanumeric value set.
   Range: DISABLE (default), ENABLE.
8. **`hptimais`** – HP Trace Identifier Mismatch AIS Insertion
   This parameter specifies if a detected TIM will cause AIS insertion. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
   Type: Alphanumeric value set.

Range: DISABLE , ENABLE.
Default: The default value depends on interface standard specified by istd for the VCG owning this tributary. For definition of *istd* refer to "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078) Output parameters. DISABLE, if *istd* = SONET; ENABLE, if *istd* = SDH.

9. **hptimmon** – HP Trace Identifier Mismatch Monitor
This parameter specifies if the mode of the TIM detection. This parameter is relevant only if the related trace read format *ptrcrfmt* is set to 16 (16-byte-specific-string mode).
Type: Alphanumeric value set.
Range: DISABLE (default), ENABLE.

10. **inmbs** – Input Membership
This parameter specifies if the tributary shall take part to the transmission of its owning VCG in input direction. Input direction in this case means from the LCAS source on the GE1 pack to the cross-connect pack XC***n***. The addressing scheme of the tributaries by using VCGNumAIDs is ensuring that only legal AIDs are used. So no consideration about the "illegal" tributaries (for example, 1-1-#-#-1-1-3 for VC-4) is necessary. The parameter *inmbs* is only valid for modifiers different from STST and VCT, that means, is omitted in the command response for STST and VCT.
The system shall not evaluate the parameter *inmbs* if the parameter *vcat* of the related VCG is set to DISABLE. The parameter is retrievable.
Type: Alphanumeric value set.
Range: DISABLE , ENABLE.
Default:
    - for modifiers STST or VCT: ENABLE
    - for all other modifiers: DISABLE.

    The VCG for GE***n*** and ES64U has as default the size 0, that means, it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user decide explicitly if one tributary take part to the transmission.

11. **inmbst** – Input Member Status
This parameter reports the status of the tributary related to its VCG membership in input direction. The parameter *inmbst* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
Type: Alphanumeric value set.
Range:
    - for modifiers STST or VCT always ACTIVE is reported
    - for all other modifiers:
    - NA, for not applicable, i.e. if the trib. is no member of the VCG or if virtual concatenation is disabled
    - ADD, the tributary is in the LCAS state ″Wait-for-Add″, ″Add″ or ″Wait-for-ACK″

- ACTIVE, the tributary is member of the VCG and working
- FAILED, the tributary is member of the VCG but taken off by LCAS (incoming failed condition for the sink tributary)
- UNAVAILABLE, for not available, i.e. the trib. is member of the VCG but the current data is not retrievable for any reason.

  The state FAILED can only be reported if *lcasmd* is set to ENABLE for the VCG owning this tributary. For parameter *lcasmd* refer to the Output parameters section of "RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078).

  For VCG tributaries located on ES64U packs only ″NA″ will be reported.

12. **`neppm`** – Near End Path PM Enable
    This parameter specifies whether the performance monitoring for the near end is enabled or disabled for this tributary. The parameter neppm is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs this parameter cannot be modified but will be reported with default values since performance monitoring on VCGTRIB level is not supported.
    Type: Alphanumeric value set.

    Range:

    - DISABLE (default)
    - ENABLE.

13. **`outmbs`** – Output Membership
    This parameter specifies if the tributary shall take part to the transmission of its owning VCG in input direction. Output direction in this case means from the cross-connect pack XCn to the LCAS sink on the GE1 pack. The parameter *outmbs* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    The system shall not evaluate the parameter outmbs if the parameter vcat of the related VCG is set to DISABLE. The parameter is retrievable.
    Type: Alphanumeric value set.

    Range:

    - DISABLE
    - ENABLE.

    Default: for modifiers STST or VCT: ENABLE. for all other modifiers:DISABLE.
    The VCG for GE***n*** and ES64U has as default the size 0, i.e. it has no member. So the default for the parameter above for VCGTRIB needs to be DISABLE so that the user decide explicitly if one tributary take part to the transmission.

14. **`outmbst`** – Output Member Status
    This parameter reports the status of the tributary related to its VCG membership in output direction. The parameter *outmbst* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    Type: Alphanumeric value set.

Range:

- for modifiers STST or VCT always ACTIVE is reported, for all other modifiers:
- NA, for not applicable, i.e. if the trib. is no member of the VCG or if virtual concatenation is disabled
- ADD, the tributary is in the LCAS state ″Wait-for-Add″ or ″Wait-for-Data″
- ACTIVE, the tributary is member of the VCG and working
- FAILED, the tributary is member of the VCG but not working due to a failure condition. Includes period during ″Hold-Off″.
- WTR, failure condition has cleared, tributary is in wait to restore condition.
- UNAVAILABLE, for not available, i.e. the trib. is member status is not retrievable for any reason.

  The state WTR can only be reported if lcasmd is set to ENABLE for the VCG owning this tributary. For parameter lcasmd refer to RTRV-VCG Output Parameters. For a detailed explanation of the ADD state in the LCAS protection scheme, please refer to SRD_DATA_Virtual_Concatenation.

  For VCG tributaries located on ES64U packs only ″NA″ will be reported.

15. **pmhpfeses** – PM HP Far End SES Threshold
    This parameter specifies the Higher Order Path Far End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the far end. The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). If *istd* is not set to SDH then *pmhpfeses* has no affect to NE behavior. The parameter *pmhpfeses* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported..
    Type: Integer.
    Range: 1, 2, 3, ..., 63998, 63999, 64000, includes SDH/SONET range.
    Default: 2400.

16. **pmhpneses** – PM HP Near End SES Threshold
    This parameter specifies the Higher Order Path Near End Severely Errored Seconds Performance Monitoring Threshold for errored frames. If errored frames are crossing this threshold this will result in an SES count for the near end.
    The behavior depends on interface standard specified by *istd* for the VCG owning this tributary. For definition of *istd* refer to the output parameters section of “RTRV-VCG: Retrieve virtual concatenation group” (p. 3-3078). If *istd* is not set to SDH then *pmhpneses* has no affect to NE behavior.
    The parameter pmhpneses is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
    For VCG tributaries located on ES64U packs this parameter can be modified

although it has currently no effect as performance monitoring on VCGTRIB level is not supported..
Type: Integer.
Range: 1, 2, 3, …, 63998, 63999, 64000, includes SDH/SONET range.
Default: 2400.

17. **`ptca`** – Path TCA Profile Pointer
This parameter specifies the name of the threshold crossing alert (TCA) profile used for this tributary.
The parameter *ptca* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported..
Type: Alphanumeric string.
Range: 1 … 24 characters.
Default: DEFAULT0.

18. **`ptpn`** – Path Terminating ASA Profile Name
This parameter points to a tributary ASAP profile name pfname for the designated AID.
Type: Alphanumeric string. A hyphen (“-”) is allowed in the name additionally.
Range: 1 … 24 characters.
Default: DEFAULT.

19. **`ptrc`** – Incoming Path Trace
This parameter reports the accepted higher order path trace identifier (J1). The length depends on the value of the parameter *ptrcrfmt*. The status of this value is contained in parameter *ptrcst*. If *ptrcst* indicates NORMAL, then *ptrc* contains a valid value. If *ptrcst* indicates UNAVAILABLE or AIS, then no valid incoming path trace identifier could be obtained by the system, e.g. due to not meeting the basic acceptance criteria or intra-NE communication problems, and the output parameter *ptrc* is omitted from the response.
Type: Hexadecimal.

    Range: the range depends on the parameter ptrcrfmt specified in RTRV-VCGTRIB Output Parameters

    - 2 hexadecimal digits, if ptrcrfmt=2
    - 32 hexadecimal digits, if ptrcrfmt=16
    - 128 hexadecimal digits, if ptrcrfmt=64.

20. **`ptrcrfmt`** – Path Trace Read Format
This parameter specifies the format of the path trace (J1 byte) to be received (sink).
For VCG tributaries located on ES64U packs this parameter is visible with default values for retrieval, but cannot be modified.
Type: numeric string.

Range:

- 2, for non-specific repeating byte
- 16, for 16 byte specific string
- 64, for 64 byte string.

Default:
For VCG tributaries located on GE1 packs: 2
For VCG tributaries located on GE10PL1 or ES64U packs: 16 if istd = SDH, 64 if istd = SONET.

21. **`ptrcst`** – Status of Incoming Path Trace
This parameter shows the status of the value of the incoming Path Trace Identifier (J1) which is available with parameter *ptrc*. For VCG tributaries located on ES64U packs this parameter is always reported with UNAVAILABLE.
Type: Alphanumeric value set.
Range:
    - NORMAL, *ptrc* contains a valid value
    - UNAVAILABLE, no valid value for *ptrc* could be obtained by the system
    - AIS , no valid value for ptrc could be obtained due to SF condition.

22. **`ptrcwfmt`** – Path Trace Write Format
This parameter specifies the format of the path trace (J1 byte) to be sent (source). For VCG tributaries located on ES64U packs this parameter is visible with default values for retrieval, but cannot be modified.
Type: numeric string.
Range:
    - 2, for non-specific repeating byte
    - 16, for 16 byte specific string
    - 64, for 64 byte string.

    Default:
For VCG tributaries located on GE1 packs: 2
For VCG tributaries located on GE10PL1 or ES64U packs: 16 if istd = SDH, 64 if istd = SONET.

23. **`rsiglb`** – Received Signal Label
This parameter reports the accepted trail signal label (C2). The parameter is not reported if the *trailst* parameter indicates a fail condition, i.e. has the value AIS or UNAVAILABLE.
Type: Alphanumeric value set.
Range: 00, 01, …, FE, FF, i.e. two hexadecimal digits.

24. **`sdsfmode`** – Signal Degrade/Signal Fail Mode
This parameter specifies the determination mode of degraded defect for transmission. The parameter sdsfmode is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
Type: Alphanumeric value set.

Range: BURST or POISSON.
Default: For VCG tributaries located on a GE10PL1 pack, the default is: BURST. For VCG tributaries located on other packs, the default value depends on interface standard specified by istd for the VCG owning this tributary. For definition of istd refer to RTRV-VCG Output Parameters. POISSON, if *istd* = SONET; BURST, if *istd* = SDH.

25. **`tcamd`** – TCA mode
This parameter is only available for SDH modeand the modifier must be VC. The parameter shows the modus of the PM thresholding. Two thresholding methods are applicable for SDH: TR-only (also known as the implicit reset method), TR-RTR (also known as the explicit reset method). For SONET the parameter *tcamd* does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
Type: Alphanumeric value set.
Range: TR, TRRTR.
Default: SDH mode: TRRTR. SONET mode: TR (internal). For SONET the parameter tcamd does not exist. SONET ports only support ″TR-only″ mode. Thus the mode must be internally set to ″TR-only″.
The parameter *tcamd* is only valid for modifiers different from STST and VCT, i.e. is omitted in the command response for STST and VCT.
For VCG tributaries located on ES64U packs this parameter can be modified although it has currently no effect as performance monitoring on VCGTRIB level is not supported.

26. **`tmonmd`** – Tributary Monitoring Mode
This parameter specifies the monitoring mode for the tributary. For VCG tributaries located on ES64U packs this parameter can be modified although it has currently no effect.
Type: Alphanumeric value set.
Range: MON or NMON (default).

27. **`trailst`** – Trail Status
This parameter reports the accepted trail status value.
Type: Alphanumeric string.
Range:
    - AIS, for Alarm Indication Signal in case of Trail Signal Fail Condition
    - UNAVAILABLE, for unavailable
    - NORMAL.

28. **`trc`** – Outgoing Path Trace
    This parameter specifies the path trace (J1) to be sent (source). For the 16 byte path trace format the following syntactical rule is applied: The first byte value is in the range from 80 to FF (MSB always set to 1). The other byte values are in the range from 00 to 7F (MSB always set to 0). This parameter is irrelevant if the format is non-specific repeating byte (*ptrcwfmt* = 2).
    For VCG tributaries located on ES64U packs, this parameter shall be visible with default values for retrieval, but cannot be modified.
    Type: Hexadecimal number without a preceding "H".

    Range: the range depends on the parameter ptrcwfmt specified in **`RTRV-VCGTRIB`** Output parameters length

    - 32, hexadecimal digits, if *ptrcwfmt* = 16
    - 128, hexadecimal digits, if *ptrcwfmt* = 64
    - value: specific string acc. ITU/ETSI mode 1, if *ptrcwfmt* = 16; 64 byte string, if *ptrcwfmt* = 16.

    Default: Depending on the default value of *ptrcwfmt*:

    - 00hex, if *ptrcwfmt* = 2.
    - 89hex followed by 15 bytes 00hex, if *ptrcwfmt* = 16.
    - 64 bytes 00hex, if *ptrcwfmt* = 64.

29. **`tslmmon`** – Trail Signal Label Mismatch Monitor
    This parameter specifies the mode of the TSL mismatch detection.
    For VCG tributaries located on ES64U packs, this parameter is visible with default values for retrieval, but cannot be modified.
    Type: Alphanumeric value set.

    Range:

    - DISABLE
    - ENABLE (default).

Detailed behavior description

The **`RTRV-VCGTRIB`** does not generate a **`REPT DBCHG`** message. The command retrieves all port parameter as described in section Output parameters.

Dependencies

brstintvl and *brstthr* are only meaningful if *sdfsmode* is set to BURST. *degthr* and *dexcthr* are only meaningful if *sdfsmode* is set to POISSON.

Parameters which are not meaningful at the moment are reported with their default values, nevertheless.

Example input/output

```
RTRV-VCGTRIB-STS:LT-UNITE:1-1-#-#-1-V2-4:123456;
```

```
   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-1-v2-4::aseqno=2,tseqno=2″
   ″1-1-#-#-1-v2-4,STS::brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6″,dexcthr=-
      3,hptimais=ENABLE,exphptrc=890000000000000000000000000000,hptimmon=DISABLE,inmbs=DISABL
;


RTRV-VCGTRIB-STS:LT-UNITE:1-1-#-#-1-V2-all:123456;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-1-v2-1,STS::brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6″,dexcthr=-
      3,hptimais=ENABLE,exphptrc=890000000000000000000000000000,hptimmon=DISABLE,inmbs=DISABL
   ″1-1-#-#-1-v2-3,STS::brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6″,dexcthr=-
      3,hptimais=ENABLE,exphptrc=890000000000000000000000000000,hptimmon=DISABLE,inmbs=DISABL

<listing all VCGTRIB’s of this VCG>

   ″1-1-#-#-1-v2-21,STS::brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6″,dexcthr=-
      3,hptimais=ENABLE,exphptrc=890000000000000000000000000000,hptimmon=DISABLE,inmbs=DISABL
;
```

Retrieval of VCGTRIB entities for a 10 GE interface on the GE10PL1 pack (Ethernet signal mapped into a contiguous concatenated signal):

```
RTRV-VCGTRIB-VC:LT-UNITE:1-1-#-#-1-v9-all:123456:::dyn=YES;

   LT-UNITE 00-11-11 11:11:00
M  123456 COMPLD
   ″1-1-#-#-1-v9-1,VC::\
    brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-3,\
    hptimais=ENABLE,exphptrc=890000000000000000000000000000,\
    hptimmon=DISABLE,inmbs=DISABLE,inmbst=NA,outmbs=DISABLE,\
    outmbst=NA,ptpn=DEFAULT,\
    ptrc=890000000000000000000000000000,ptrcrfmt=16,\
    ptrc=NORMAL,ptrcwfmt=16,rsiglb=00,sdsfmode=BURST,\
    tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE″
   ″1-1-#-#-1-v9-4,VC::\
    brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-3,\
    hptimais=ENABLE,exphptrc=890000000000000000000000000000,\
    hptimmon=DISABLE,inmbs=ENABLE,inmbst=FAILED,outmbs=ENABLE,\
    outmbst=FAILED,ptpn=DEFAULT,\
    ptrc=890000000000000000000000000000,ptrcrfmt=16,\
    ptrc=NORMAL,ptrcwfmt=16,rsiglb=00,sdsfmode=BURST,\
    tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE″

<listing all VCGTRIB’s of this VCG>
```

```
   "1-1-#-#-1-v9-190,VC::\
    brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr=-3,\
    hptimais=ENABLE,exphptrc=890000000000000000000000000000000,\
    hptimmon=DISABLE,inmbs=ENABLE,inmbst=FAILED,outmbs=ENABLE,\
    outmbst=FAILED,ptpn=DEFAULT,\
    ptrc=890000000000000000000000000000000,ptrcrfmt=16,\
    ptrc=NORMAL,ptrcwfmt=16,rsiglb=00,sdsfmode=BURST,\
    tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"
;
```

Retrieval of VCGTRIB entities for a VCG on the ES64U pack:

```
RTRV-VCGTRIB-STS:UNITE-NE:1-1-#-#-22-v9-all:123456:::dyn=YES;

  LT-UNITE 00-11-11 11:11:00
A  001 REPT DBCHG
   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-1::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"

   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-2::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"


   "TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-3::\
   brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
   feppm=DISABLE, hptimais=DISABLE,\
   exphptrc=890000000000000000000000000000000,hptimmon=DISABLE,\
   inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
   outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
   ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
   ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
   tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"


   <listing all VCGTRIB's of this VCG>
```

```
"TIME=11-10-30,DATE=00-11-11:RTRV-VCGTRIB-STS:1-1-#-#-22-v9-12::\
brstintvl=7,brstthr=30,crslbkstat=NO,degthr=-6,dexcthr= 3,\
feppm=DISABLE, hptimais=DISABLE,\
exphptrc=89000000000000000000000000000000,hptimmon=DISABLE,\
inmbs=DISABLE,inmbst=NAV,neppm=DISABLE,outmbs=DISABLE,\
outmbst=NAV,pmhpfeses=2400,pmhpneses=2400,ptca=DEFAULT0 ,\
ptpn=DEFAULT,ptrc=64,ptrcst=UNAVAILABLE,ptrcrfmt=64,\
ptrcwfmt=64,rsiglb=01,sdsfmode=BURST,tcamd=TR,\
tmonmd=NMON,trailst=NORMAL,trc=00,tslmmon=ENABLE"

;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-VCGTRIB** command, too.

Validation rule: Check *modifier* against circuit pack type

If the command is applied

- with modifier = VC or STS to a VCG tributary on a transparent pack or
- with modifier = VCT or STST to a VCG Tributary on a GE***n*** or ES64U pack, then the command is denied.

Error code: `IENE – Input, Entity Not Exists`

Validation rule: Check CCB against *istd* of related VCG

Check wether the CCB fits to *istd* of the related VCG.

Error code:
`ICNV – Input, Command Not Valid, Interface Standard does not match`.

E.g. DENY, if a command with *modifier* = VC or *modifier* = VCT is made on a port which is provisioned as *istd* = SONET.

Related TL1 messages

- “ED-VCGTRIB: Edit virtual concatenation group tributary” (p. 3-1416)
- “OPR-VCGTRIB: Operate virtual concatenation group tributary” (p. 3-2082)

RTRV-VRTSW: Retrieve virtual switch

The **RTRV-VRTSW** command retrieves the attributes of a virtual switch of a GE1 pack.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): P1

Priority: 1

Abortable: Yes

Input format

Command syntax: **RTRV-VRTSW**:*tid*:*aid*:*ctag[:::spec_block]*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   Refer to the "RTRV-HDR: Retrieve header" (p. 3-2417) command for the input parameter syntax and the description of this parameter.
2. **aid** – Access Identifier
   The access identifier specifies the slot containing a GE1 pack.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure.
   The values
   - all and
   - usall for Simple_UnivSlotAID

   is accepted.
3. **ctag** – Correlation Tag
   See definition of *ctag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. **dyn** – Retrieve Dynamic Parameters
   This parameter controls whether dynamic parameters are reported with the command reply or not. It is given only when using any of the wildcard values all or usall at the appropriate positions of the input AID. For input AIDs not using *all* or *usall* this parameter is not allowed. This implies that with such input AIDs the dynamic parameters are always returned.
   The following output parameters of a virtual switch are affected:
   - *macaddr*
   - *rootpriority*
   - *rootmacaddr*

- *rootcost*
- *rootport*.

Type: Alphanumeric value set.
Range: YES, report dynamic parameters in the reply as well. NO (default), do not report dynamic parameters in the reply.

Output format

After receiving the **RTRV-VRTSW** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
   ”aid:vrtswid:spec_block”
   <this entire block is repeated for another vrtswid>
   <this entire structure is repeated for another slot>
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of.

The output is in the form of a nested list. The primary list is based on the slot AID. For each slot AID, there is a nested list for each instance of a virtual switch giving the spec_block parameters.

Output parameters

Command parameters

1. **aid** – Access Identifier
   This parameter specifies the slot containing a GE1 pack to be operated on.
   Type: AID.
   Range: UnivSlotAID as defined in AID structure with the exception of the value all and usall.
2. **vrtswid** – Virtual Switch Identifier
   This parameter identifiers the instance of the virtual switch.
   - A Virtual Switch ID is unique within a single circuit pack. When creating a new Virtual Switch, the system automatically assigns a unique ID to it.
   - There will be one virtual switch per physical switch, implicitly all LAN ports and all WAN ports on the physical switch are part of the single virtual switch. The ID of this virtual switch is fixed to 1. A future release may support multiple virtual switches per physical switch.

   Type: Integer.
   Range: 1 – Currently this is the only allowed value.
   Default: 1.

Spec block parameters

1. **`eplst`** – EPORT AID list
   This parameter specifies a list of EPORT instances connected to the virtual switch instance as specified by *vrtswid*.
   In R3 there is only one virtual switch instance to which all ports are mapped. Therefore this list is fix.
   Type: AID list.
   Range: A list of EPortAIDs as specified in the AID definition section with the exception of the values all and usall in the respective components of the AID, separated by an ampersand ('&')
   Example: If all four EPORTs of a GE1 are mapped to one virtual switch:
   1-#-#-1-1-1&1-#-#-1-1-2&1-#-#-1-1-3&1-#-#-1-1-4
2. **`ethertype`** – VLAN tag protocol ID
   This parameter specifies the VLAN tag protocol ID necessary for interworking purposes. This parameter is used to control whether the virtual switch operates in IEEE802.1Q tagging mode (i.e. set to 0x8100) or in transparent tagging mode (i.e. set to value different to 0x8100).
   Type: Hexadecimal value.
   Range: 0x0600 … 0xFFFF.

   The following values have a specific meaning:

   - 0x8100: virtual switch operates in IEEE802.1Q tagging mode
   - any other value: virtual switch operates in transparent tagging mode.

   Default: 0x8100.
3. **`gvrpmode`** – GVRP mode
   The parameter *gvrpmode* allows to control the dynamic VLAN registration mode.
   Type: Alphanumeric value set.
   Range: ENABLED, GVRP is enabled. DISABLED (default), GVRP is disabled.
4. **`macaddr`** – Bridge ID address
   The parameter *macaddr* specifies the bridge address field of the bridge identifier.
   Type: Alphanumeric value set.
   Range: 000000000000 to FFFFFFFFFFFF (hexadecimal value) and UNAVAILABLE, if the value can not be retrieved.
5. **`rootcost`** – Cost of Path to root bridge
   This parameter specifies the cost of the path to the root bridge as determined by the *Spanning Tree Protocol*.
   Type: Alphanumeric value set.
   Range: 0 to 4 294 967 295 and UNAVAILABLE if the value can not be retrieved.
6. **`rootmacaddr`** – Root bridge ID address
   This parameter specifies the bridge identifier of the root bridge as determined by the *Spanning Tree Protocol*.
   Type: Alphanumeric value set.
   Range: 000000000000 to FFFFFFFFFFFF (hexadecimal value) and UNAVAILABLE if the value can not be retrieved.

7. **`rootport`** – Root port identifier
   This parameter specifies the identifier of the root port, which offers the lowest cost path from the virtual switch to the root bridge.
   Type: AID or alphanumeric string.
   Range: *EPortAID* or *VCGAID* as specified in the AID definition section with the exception of the values *all*, *usall* and *vall* in the respective components of the AID; UNAVAILABLE.
   The value UNAVAILABLE is returned when the addressed virtual switch is currently the spanning tree root bridge. This is also the case if STP is disabled for all ports belonging to the virtual switch. The Value UNAVAILABLE is returned if the value can not be retrieved from the HW.
8. **`rootpriority`** – Root bridge priority
   The parameter *rootpriority* specifies the priority of the root bridge as determined by the Spanning Tree Protocol.
   The virtual switch defined with the lowest value for its bridge priority compared to other virtual switches in the network will be elected as the root bridge.
   Type: AID or alphanumeric string.
   Range: 0 to 65535 and UNAVAILABLE if the value can not be retrieved.
   Default: 32678.
9. **`rstpmd`** – Rapid Spanning Tree Protocol Mode
   This parameter allows to enable or disable the Rapid Spanning Tree Protocol for the given virtual switch.
   Type: Alphanumeric value set.
   Range: DISABLE (rSTP is disabled, i.e. the virtual switch only supports normal STP (IEEE802.1D)); ENABLE (rSTP is enabled, i.e. the virtual switch also supports rSTP (IEEE802.1w)).
   Default: ENABLE
10. **`stbrpriority`** – Spanning tree bridge priority
    This parameter specifies the priority of the bridge to be used by the Spanning Tree Protocol.
    Type: Integer.
    Range: 0 to 65535.
    The stepping as recommended by the standard (steps of 4096 starting at 0) should be implemented by the management system. The NE will accept any value in the range described above. This has the consequence, that also the range restriction ″0 to 61440″ is not done by the NE.
    Default: 32768.
11. **`stfdtm`** – Spanning Tree forward delay timer
    The parameter *stfdtm* specifies the rSTP-related bridge forward delay timer. This parameter is relevant for both rSTP and STP.
    Type: Integer.
    Range: 1 … 255.
    Default: 15.

12. **sthellotm** – Spanning Tree hello timer
 The parameter *sthellotm* specifies the rSTP-related bridge hello timer. This parameter is relevant for both rSTP and STP.
 Type: Integer.
 Range: 1 … 225.
 Default: 2.
13. **stmaxage** – Spanning Tree maximum age timer
 The parameter *stmaxage* specifies the rSTP-related maximum PDU transmission delay timer. This parameter is relevant for both rSTP and STP.
 Type: Integer.
 Range: 1 … 255.
 Default: 20.
14. **vcglst** – VCG AID list
 The parameter *vcglst* specifies a list of VCG instances connected to the virtual switch instance as specified by *vrtswid*.
 In R3 there is only one virtual switch instance to which all eports are mapped. Therefore this list is fix.
 Type: AID list.
 Range: A list of VCGAIDs as specified in the AID definition section with the exception of the values all, usall and vall in the respective components of the AID, separated by an ampersand ('&').
 Example: If all four VCGs of a GE1 are mapped to one virtual switch:
 1-#-#-1-1-v1&1-#-#-1-1-v2&1-#-#-1-1-v3&1-#-#-1-1-v4.

Detailed behavior description

Omit dynamic parameters if not accessible

If the system cannot access dynamic data related to the addressed slot, e.g. due to a missing or defect pack or a system-internal communication outage, the following parameters shall response the value ″UNAVAILABLE″::

- *macaddr*
- *rootpriority*
- *rootmacaddr*
- *rootcost*
- *rootport*.

Example input/output

```
RTRV-VRTSW:LT-UNITE:1-1-#-#-1:123456;
   LT-UNITE 02-05-15 13:31:30
M  123456 COMPLD
   ″1-1-#-#-1:1:eplst=1-1-#-#-1-1&1-1-#-#-1-2&1-1-#-#-1-3&1-1-#-#-1-
      4,gvrpmode=ENABLE,macaddr=010203040506,rootcost=1234567890,rootmacaddr=112233445566,ro
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-EPORT** command.

Related TL1 messages

- “ED-VRTSW: Edit virtual switch” (p. 3-1439)

Security event notification

Output format

Syntax:

REPT EVT *modifier aid:condtype,condeff,ocrdat,ocrtm:conddescr;*

Output parameters

Command parameters

1. **`modifier`** – Modifier
   The parameter specifies the value ’COM’ as an event that applies to the whole NE.
   Type: Alphanumeric value set.
   Range: COM
2. **`aid`** – Access Identifier
   Type: AID
   Range: SystemAID
3. **`condtype`** – Condition Type
   This parameter specifies/identifies the type of security event being reported.
   Type: Alphanumeric value set.
   Range:
   - SEC-USERLOGIN: Successful user login
4. **`condeff`** – Condition Effect
   The parameter shall indicate the effect of the reported security event on the NE.
   Type: Alphanumeric value set.
   Range:
   - SC: Standing condition
   - CL: Clear
5. **`ocrdat`** – Occurrence Date
   The parameter shall indicate the date of the security event being reported.
   Type: value set of format YY-MM-DD
   Range: Y, M, D are digits. Leading zeros are not suppressed. YY represents the year, MM the month, DD the day.
   Default: None.

6. **`ocrtm`** – Occurrence Time
   The parameter shall indicate the time of the security event being reported.
   Type: value set of format HH-MM-SS
   Range: H, M, S are digits. Leading zeros are not suppressed. HH represents hours, MM minutes, SS seconds.
   Default: None.

7. **`conddescr`** – Condition Description
   The parameter shall provide additional information w.r.t. the security event.
   Type: inner string
   Range:
   The following strings are applicable for the one remaining *condtype* SEC-USERLOGIN.
   Also see the relation between *condeff* and *conddescr*:
   condeff – *conddescr*
   SC – \″login – <uid>\″
   CL – \″logout – <uid>\″
   CL – \″forced logout – <uid> \″
   CL – \″inactivity timeout – \″
   CL – \″logout, disconnected – <uid> \″
   The security event notification is related to the user session which is indicated by the user ID (*uid*).

Example outputs

```
   sid date time
M  ctag COMPLD
;

   sid date time
M  ctag DENY
   SROF
   /* Status, Requested Operation Failed, processing failure */
;
```

SET-ATTR-CONT: Set attribute control

The **SET-ATTR-CONT** command can be initiated by users to set the description associated with an MDO port (Miscellaneous Discrete Output / external discrete control).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No.

Input format

Command syntax: **SET-ATTR-CONT**:*tid*:*aid*:*ctag*::*contmsg[,,asgcond]*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – Access Identifier
   This is the address of the MDO port entities on which the command operates. If misc_outall is specified the configuration of all MDO's will be retrieved.
   Type: AID
   Range: Miscellaneous_out_discreteAID
   Default: misc_outall (if the *aid* is omitted).
3. **ctag** – Correlation Tag
   See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
4. **contmsg** – Control Message
   This is the name associated to a MDO port (environmental control).
   Type: text string
   Range: A string of up to 26 characters, any character except ″ and \, upper and lower cases considered. An empty string is allowed. The value would be ″″.
5. **asgncond** – Assigned Condition Type
   This parameter identifies the probable cause (type of alarm indication) which is assigned to an MDO port (external control). If one or more alarm(s) of this probable cause appears then the MDO port (external control) will be switched on.
   Type: Alphanumeric value set. Additionally a "-" will be allowed (not the first character).
   Range: All defined probable causes and UNDEF: if no probable cause is assigned.
   Default: UNDEF.
   If the value is omitted, then the actual value is not changed.

Output format

The following example shows the successful completion of the **SET-ATTR-CONT** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section "Output parameters" of "RTRV-HDR: Retrieve header" (p. 3-2417).

Detailed behavior description

The name of a MDO via the aid specified will be updated according the value of *contmsg*. The update will be reported via a SET-ATTR-CONT – REPT DBCHG message. Refer to the definition of the SET-ATTR-CONT notification.

Example input/output

The following example shows the successful completion of the **SET-ATTR-CONT** command by the NE:

```
SET-ATTR-CONT:LT-UNITE-123:MISC_OUT3:123456::"fan 3",,FANF;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

The following example shows the successful completion of the **SET-ATTR-CONT** command by the NE if the *asgncond* parameter will be unchanged:

```
SET-ATTR-CONT:LT-UNITE-123:MISC_OUT3:123456::"fan 3";

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section "Error responses" of "RTRV-HDR: Retrieve header" (p. 3-2417). The error responses listed there also apply to the **SET-ATTR-CONT** command.

Related TL1 messages

- "RTRV-ACTL-CONT: Retrieve actual condition types" (p. 3-2219)
- "RTRV-ATTR-CONT: Retrieve attribute control" (p. 3-2254)

- "OPR-EXT-CONT: Operate external control" (p. 3-2032)
- "RLS-EXT-CONT: Release external control" (p. 3-2190).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

SET-ATTR-CONT notification

Output format

The **SET-ATTR-CONT** notification has the following specific syntax:

SET-ATTR-CONT:*aid::contmsg,,asgncond*

Output parameters

Command parameters

1. **aid** – Access Identifier
   This is the address of the MDO port entity for which the configuration is retrieved or on which an operation is performed.
   Type: AID.
   Range: Miscellaneous_out_discreteAID. misc_outall must not be reported or used.
2. **contmsg** – Control Message
   This is the name associated to a MDO port (environmental control). It is a description which can be retrieved by the management system.
   Type: inner string.
   Range: string up to 26 characters, any character except ″ and \, upper and lower cases considered. A empty string is allowed. The value would be \″\″.
   Default: MDO port #n. n is the number of the port 1 … 8.
3. **asgncond** – Assigned Condition Type
   This parameter identifies the probable cause (type of alarm indication) which is assigned to an MDO port (external control). If one or more alarm(s) of this probable cause appears then the MDO port (external control) will be switched on.
   Type: Alphanumeric value set. Additionally a “-” will be allowed (not the first character).
   Range: All defined probable causes and UNDEF: if no probable cause is assigned.
   Default: UNDEF.

Example output

The example shows a normal SET-ATTR-CONT notification:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ″TIME=08-00-12,DATE=00-01-01:SET-ATTR-CONT:MISC_OUT3::\″fan 3\″,,FANF″
;
```

SET-ATTR-ENV: Set attribute environment

The **SET-ATTR-ENV** command can be used to set the alarm message, name and assigned ASAP of an miscellaneous discrete input (MDI/environmental point).

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **SET-ALM-ENV**:*tid*:*aid*:*ctag*::,,*almmsg*,*envname*:*spec_block*;

Input parameters

Command parameters

- **tid** – Target Identifier
  See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).
- **aid** – Access Identifier
  This is the address of the MDI port entities for which an alarm condition or the configuration is being reported. If misc_inall is specified the information for all entities will be retrieved.
  Type: AID.
  Range: Miscellaneous_in_discreteAID.
  Default: misc_inall (if the aid is omitted).
- **ctag** – Correlation Tag
  See definition of "RTRV-HDR: Retrieve header" (p. 3-2417).

- **almmsg** – Alarm Message
  This is the condition description which is associated with the MDI port (environmental point). This description will be sent with the alarm generated from the MDI port.
  Type: inner string.
  Range: a string up to 32 characters, any character except ″ and \, upper and lower cases are considered.
  Default: Miscellaneous Discrete #<N> with N = 1 … 8 for the different MDI ports.
- **envname** – Environmental Point Name
  This is the name associated to a MDI port (environmental point). It is a description which can be retrieved by the management system. It is not to be mixed up with the envmsg parameter. The envname is just a name of the MDI port and not reported via the condition description of an alarm.
  Type: inner string
  Range: a string up to 26 characters, any character except ″ and \, upper and lower cases considered.
  Default: MDI port #n; n is the number of the port 1 … 8.

Spec block parameters

- **envinv** – Environmental Alarm Inversion
  This parameter indicates for the MDI (environmental point) whether an alarm is generated if the port is on or if the port is off.
  Type: Alphanumeric value set.
  Range:
  – YES: The alarm will be inverted
  – NO: The alarm will be not inverted.
- **pfname** – Profile Name
  This is the name of the ASAP that is assigned to the MDI port (environmental point). The ASAP must be of the type ENV.
  Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
  Range: string up to 24 characters.
  Default: DEFAULT.

Output format

The following example shows the successful completion of the **SET-ATTR-ENV** command by the NE:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The **SET-ATTR-ENV** command updates the configuration of the almmsg, envname, pfname of the via aid specified MDI port. The changes will be reported via a REPT DBCHG message. See definition of SET-ATTR-ENV notification.

Example input/output

The following example shows the successful completion of the **SET-ATTR-ENV** command by the NE:

```
SET-ATTR-ENV:LT-UNITE-123:MISC_IN7:123456::,,”open door”,”door”:pfname=ENV_
      NM,envinv=YES;

   LT-UNITE-123 00-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **RTRV-ATTR-ENV** command.

Validation rule: *pfname* does not exist

An ASAP with the name pfname does not exist for an ASAP type ENV.

Error code: `IDNV – Input Data Not Valid, ASAP profile does not exist`

Related TL1 messages

- “ED-ASAP-PROF: Edit alarm severity assignment profile” (p. 3-284)
- “ENT-ASAP-PROF: Enter alarm severity assignment profile” (p. 3-1446)
- “REPT-ALM-ENV: Report alarm environment” (p. 3-2101)
- “RTRV-ALM-ENV: Retrieve alarm environment” (p. 3-2238)
- “RTRV-ATTR-ENV: Retrieve attribute environment” (p. 3-2257).

References

GR-833-CORE, Issue 4, June 2000, OTGR Section 12.3, Network Maintenance: Network Element and Transport Surveillance Messages.

SET-ATTR-ENV notification

Output format

The **SET-ATTR-ENV** notification has the following specific syntax:

SET-ATTR-ENV:*aid,,almmsg,envname:spec_block*;

Output parameters

Command parameters

- **aid** – Access Identifier
  This is the address of the MDI port entities for which an alarm condition or the configuration is being reported. If misc_inall is specified the information for all entities will be retrieved.
  Type: AID.
  Range: Miscellaneous_in_discreteAID.
  Default: misc_inall (if the aid is omitted).
- **almmsg** – Alarm Message
  This is the condition description which is associated with the MDI port (environmental point). This description will be sent with the alarm generated from the MDI port.
  Type: inner string.
  Range: a string up to 32 characters, any character except ″ and \, upper and lower cases are considered.
  Default: Miscellaneous Discrete #<N> with N = 1 … 8 for the different MDI ports.
- **envname** – Environmental Point Name
  This is the name associated to a MDI port (environmental point). It is a description which can be retrieved by the management system. It is not to be mixed up with the envmsg parameter. The envname is just a name of the MDI port and not reported via the condition description of an alarm.
  Type: inner string
  Range: a string up to 26 characters, any character except ″ and \, upper and lower cases considered.
  Default: MDI port #n; n is the number of the port 1 … 8.

Spec block parameters

- **envinv** – Environmental Alarm Inversion
  This parameter indicates for the MDI (environmental point) whether an alarm is generated if the port is on or if the port is off.
  Type: Alphanumeric value set.

Range:

– YES: The alarm will be inverted

– NO: The alarm will be not inverted.

- **pfname** – Profile Name
  This is the name of the ASAP that is assigned to the MDI port (environmental point). The ASAP must be of the type ENV.
  Type: alphanumeric character string. A hyphen (“-”) is allowed in the name additionally.
  Range: string up to 24 characters.
  Default: DEFAULT.

Example output

The example shows a normal notification after an SET-ATTR-ENV Command:

```
   LT-UNITE-123 00-01-01 08:00:50
A 815 REPT DBCHG
   ”TIME=08-00-12,DATE=00-01-01:SET-ATTR-ENV:MISC_IN7::,,\”open
      door\”,\”door\”:pfname=ENV_NM,envinv=YES”
;
```

SET-MAP-RING: Set map ring

The **SET-MAP-RING** command writes the transitional, manually provisioned ring map to NVM for use by the MS-SPRing/BLSR protection group.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No

Input format

Command syntax: **SET-MAP-RING**:*tid*:[*aid*]:*ctag*::[*rid*]*;*

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
3. **ctag** – Correlation Tag
   See definition in input parameters section “RTRV-HDR: Retrieve header” (p. 3-2417).
4. **rid** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: If *aid* is specified, then *rid* can be omitted.

Output format

After receiving the **SET-MAP-RING** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

Output parameters

1. **sid** – Source Identifier
   This is the currently active name of the network element to which the command is addressed. The tid is a case insensitive character string.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed.
   Range: 1 to 20 characters.
2. **date** – Date
   Date when output message is generated. This has the format YY-MM-DD (year-month-day)
   Type: value set.
   Range: for YY 00 to 99, for MM 01 to 12, for DD 01 to 31
   TL1 messages use YY-MM-DD for the format of a date, meaning that the year only has two digits. This is consistent with Telcordia requirements. Internally, however, the system does use a 4-digit year. When the year appears in a response, the internal 4-digit year is truncated to two digits. When the year appears as an input parameter, for example in the **ED-DAT** command, years in the range 90-99 will represent 1990-1999, and years in the range 00-37 will represent 2000-2037.
   Default:00-01-01
3. **time** – Time
   Time when output message is generated. This has the format HH:MM:SS (hours:minutes:seconds).
   Type: Value set.
   Range: for HH 00 to 23, for MM 00 to 59, for SS 00 to 59.
   Default: 00:00:00
4. **M** – Message
   M indicates that the output message is generated in response to a manual command.
   Type: Value set.
   Range: M

5. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: Non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.
6. **IP** – In Progress
   IP is sent in response to all commands with the exception of commands that result in error code output.
   Type: Value set.
   Range: IP
7. **RL** – Repeat Later
   The requested command cannot be executed due to unavailable system resources. The command may be entered again later.
   Type: Value set.
   Range: RL
8. **COMPLD** – Completion Code
   It indicates that the command has been completed.
   COMPLD for successful command execution: For a command that is decomposed into a set of atomic commands a COMPLD completion code means that the command was successful for all atomic commands (e.g. AID ranges, group alias).
   COMPLD for no changes: If a parameter has a current value, and a provisioning command attempts to reprovision the parameter with a value that is the same as its current value, the command will return a COMPLD completion code.
   Type: Value set.
   Range: COMPLD.

Detailed behavior description

The **SET-MAP-RING** command is used to move the transitional manually provisioned ring map to NVM foe use by the BLSR port protection group. This command will be denied if automatic ring discovery is enabled.

The **SET-MAP-RING** command is also denied if the list does not contain the local node , if the list does not contain at least 2 nodes, or if duplicate nid values appear in the list. The **SET-MAP-RING** command generates a REPT DBCHG message.

Example input/output

The following is an example of a SET-MAP-RING command:

```
SET-MAP-RING:NENAME-1::123456::rid1;
   NENAME-1 98-01-01 08:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Automatic ring discovery to be disabled

If the automatic ring discovery is enabled, the following error response is returned:

Error code: `SNVS – Status, Not Valid State`

Validation rule: consistent aid and rid

If both an aid and an rid are specified as input parameters and they are inconsistent, then the following error response is returned

Error code: `IDNV – Input, Data Not Valid, inconsistent aid and rid`

Related TL1 messages

- "ED-FETID: Edit far end target identifier" (p. 3-405)§
- "ED-MAP-RING: Edit map ring" (p. 3-436)
- "ED-PROTN-GRP: Edit protection group" (p. 3-803)
- "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)

References

GR-199-CORE

SET-MAP-RING notification

Output format

Command syntax: **`SET-MAP-RING`**:*aid[:rid];*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the aid of an MS-SPRing/BLSR protection group or an port aid being a member of the group.
   Option: If *rid* is specified, then *aid* can be omitted.
   Type: AID.
   Range: TwoFibre_ProtGrpAID (not ranged) or FourFibre_ProtGrpAID (not ranged) or SPortAID (not ranged)
2. **`rid`** – Ring Identification
   This parameter specifies the ring identification name. Storage and retrieval of the *rid* is case sensitive, but actual use of the rid to determine the corresponding protection group is case insensitive. The parameter is used as an alias to identify exactly one MS-SPRing/BLSR protection group of any type. Therefore the value is unique for all protection groups of type 2F or 4F within the NE.
   Type: text string of chars allowed for TID strings.
   Range: 1 to 15 characters.
   Default: NONE.
   Option: The rid is only included in the SET-MAP-RING notification if the rid has been specified in the **`SET-MAP-RING`** command.

Example input/output

The following example reports the creation of a ring map in LT-UNITE’s NVM:

```
LT-UNITE 00-11-03 15:00:00
A  000 REPT DBCHG
   ″TIME=15-00-00,DATE=00-11-03:SET-MAP-RING:1-1-t001:rid1″
;
```

SET-OW: Set orderwire and user bytes

General

The **SET-OW** command associates the overhead byte (E1 or E2 or F1) on an optical port to an overhead access port or removes this association. Furthermore the port-interface (either G.703 or V.11) of the overhead access port can be specified.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No.

Input format

Command syntax: **SET-OW**:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in “RTRV-HDR: Retrieve header” (p. 3-2417).
2. **aid** – Access Identifier
   The parameter *aid* specifies the overhead access port to be used for association (or removing the association).
   Type: UserBytePortAID.
   Range: As given by the UserBytePortAID definition, the value ″ubioall″ is not allowed.
3. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters. Note: Current implementation also allows “0” as valid *ctag*, which is an extension to GR-831.

Spec block parameters

1. **intfctype** – Interface Type
   This parameter specifies which electrical interface standard of the overhead access port is used.

Two different settings are possible:

- electrical interface according to G.703
- electrical interface according to V.11.

Note: The overhead access ports with AIDs ubio1, … ubio4 are configurable to either G703 or V11. The overhead access ports with AIDs ubio5, ubio6 are fixed to V11.
Type: Alphanumeric value set.
Range: G703, V11.
Default: G703 (for the overhead access ports given by the AIDs ubio1 … ubio4); V11 (for the overhead access ports given by the AIDs ubio5, ubio6 – fixed to V11).

2. **`ohbyte`** – Overhead-Byte
   This parameter specifies which of the overhead bytes is used.
   Three channels are possible:
   - E1 and F1 in the RSOH (section overhead)
   - E2 in the MSOH (line overhead).

   Type: Alphanumeric value set.
   Range: E1, F1, E2.
   Default: E1.
   Note: In output direction the empty string appears as \″\″.
3. **`portaid`** – Port Access Identifier
   This parameter specifies the optical/electrical port (of an OP***n***/EP***n***) of the userbyte to which the overhead access port (given by the command parameter aid) is associated, respectively dis-associated.
   Type: In case an association is established: SPortAID; in case an association will be removed: Alphanumeric value set (i.e. the string ″NOCONNECT″).
   Range: In case an association is established: SPortAID (No **`all`** value is supported); in case an association will be removed: NOCONNECT.
   Default: NOCONNECT.

Output format

If the **`SET-OW`** command completes successfully, then the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are described in the Output parameter section of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

The attributes *portaid*, *ohbyte* and *intfctype* of the entity Userbyte port is updated by this command.

This update is reported by a SET-OW notification.

Example:

SET-OW:UNITE-NE:ubio3:123456:::portaid=1-1-#-#-5-1,ohbyte=E2,intfctype=V11;

REPT DBCHG

SET-OW:ubio3::port_aid=1-1-#-#-5-1,ohbyte=E2,intfctype=V11;

Circuit pack type related restrictions

The Orderwire and User Byte access is only functioning on the following circuit pack types: OP155, EP155, OPLB, all OP10xx packs.

Update affected entities

Apart from the Userbyte port entity no further entities are affected by the **SET-OW** command.

Example input/output

Example 1

Suppose the Userbyte port given by AID ubio1, to be configured to V11, is associated to the E1-Byte channel of the optical port of an OP10G in slot 10:

```
SET-OW:UNITE-NE:1-1-ubio1:123456:::portaid=1-1-#-#-10-1-1,ohbyte=E1,intfctype=V11;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Example 2

Suppose the association as made in Example-1 is removed (i.e. dis-associated):

```
SET-OW:UNITE-NE:1-1-ubio1:123456:::portaid=NOCONNECT,
ohbyte=E1,intfctype=V11;


   UNITE-NE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Interface type compliant with port

Command processing shall verify that setting the interface type to G.703 is only possible for the UserByte-Ports *ubio1*, … *ubio4* (but not for ubio5 and ubio6).

Error code:
`IDNV – Input, Data Not Valid, Interface type not settable for specified UserByte-Port`

Validation rule: No association present (when associating)

Command processing shall verify that when setting up an association no association is already present.

This means, that only a one to one relation is allowed between the specified:

- Userbyte of a SDH/SONET optical/electrical port and
- UserByte port.

Error code: `SNVS – Status, Not in Valid State, association already exists`

Validation rule: Association present (when dis-associating)

Command processing shall verify that an association exists when trying to remove it.

Error code:
`IENE – Input, Entity Not Exists, Specified association does not exist`

Validation rule: *ohbyte* and *intfctype* values correct (when dis-associating)

Command processing shall verify in case of removing the association that the entered value for the parameters ohbyte and intfctype are identical with the current ones.

Error code:
`IDNV – Input, Data Not Valid, discrepancy between entered values and current value of spec-block parameters`

Validation rule: Port exists (when associating)

When setting up an association, the command processing shall verify that the specified SDH/SONET optical/electrical port exists.

Error code: `IENE – Input, Entity Not Exists, Specified port does not exist`

Validation rule: No Userbyte access at EC1 ports on EP51 pack

When setting up an association, the command processing shall verify that the specified port is not a EC1 port on an EP51 pack.

Error code:
`IDNV – Input, Data Not Valid, No Userbyte access at the specified port`

Related TL1 messages

- "RTRV-OW: Retrieve orderwire and user bytes" (p. 3-2694)

SET-OW notification

General

Command syntax: **`SET-OW`***:aid::spec-block;*

Output parameters

Command parameters

1. **`aid`** – Access Identifier
   This parameter specifies the overhead access port for which the following spec_block parameters are returned.
   Type: UserBytePortAID.
   Range: As given by the UserBytePortAID definition, the value ″ubioall″ is not allowed.

Spec block parameters

1. **`intfctype`** – Interface Type
   This parameter specifies which electrical interface standard of the overhead access port is used.

   Two different settings are possible:

   - electrical interface according to G.703
   - electrical interface according to V.11.

   Note: The overhead access ports with AIDs ubio1, … ubio4 are configurable to either G703 or V11. The overhead access ports with AIDs ubio5, ubio6 are fixed to V11.
   Type: Alphanumeric value set.
   Range: G703, V11.
   Default: G703 (for the overhead access ports given by the AIDs ubio1 … ubio4); V11 (for the overhead access ports given by the AIDs ubio5, ubio6 – fixed to V11).
2. **`ohbyte`** – Overhead-Byte
   This parameter specifies which of the overhead bytes is used.

Three channels are possible:

- E1 and F1 in the RSOH (section overhead)
- E2 in the MSOH (line overhead).

Type: Alphanumeric value set.
Range: E1, F1, E2.
Default: E1.
Note: In output direction the empty string appears as \″\″.

3. **`portaid`** – Port Access Identifier
   This parameter specifies the optical/electrical port (of an OP***n***/EP***n***) of the userbyte to which the overhead access port (given by the command parameter aid) is associated, respectively dis-associated.
   Type: In case an association is established: SPortAID; in case an association will be removed: Alphanumeric value set (i.e. the string ″NOCONNECT″).
   Range: In case an association is established: SPortAID (No **`all`** value is supported); in case an association will be removed: NOCONNECT.
   Default: NOCONNECT.

Behavior description

Refer to the *Detailed behavior description* of “SET-OW: Set orderwire and user bytes” (p. 3-3147).

SET-SID: Set SID

The **SET-SID** command is used to change the target/source identification of the NE. The command is denied if the system is not in maintenance condition.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): S4

Priority: 1

Abortable: no

Input format

Command syntax: **SET-SID***:tid::ctag::newsid*;

Input parameters

Command parameters

1. **tid** – Target Identifier
   See definition in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **ctag** – Correlation Tag
   This is included in the command by the TL1 user and is repeated by the network element in the response to allow the TL1 user to associate the command and response messages.
   Type: Alphanumeric string
   Range: non-zero decimal number or identifier of up to 6 characters.
   Note: Current implementation also allows "0" as valid *ctag*, which is an extension to GR-831.
3. **newsid** – New Target/Source Identification
   This is the new name of the NE. Future commands must use this name instead of the current name.
   Type: identifier, in addition the special character underscore "_" and the password characters are allowed, except the characters plus sign (″+″), number sign (″#″) and percent sign (″%″).
   Range: 1 to 20 characters.

Output format

If the NE fully complies with the SET-SID request, the normal completion response is returned as shown below:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

This command provides a new name to the NE.

The update is reported by means of an SET-SID notification.

The **SET-SID** command is denied if the system is not in maintenance condition. Maintenance condition applies only to NE.

The new SID will come into effect only after reset of the CTL. This reset takes place autonomously when the maintenance condition is left. Commands after the reset must use the new SID instead of the current name.

The response as well as corresponding **REPT DBCHG** notification use the old SID in the *sid* parameter.

The **SET-SID** command generates a **REPT DBCHG** message.

Updated affected entities

There is no further entities affected.

If the NE is a node in a BLSR ring/MS-SPRing and the TID is changed, then this may impact cross-connection entities of ring neighbors. Since these updates are reported by NEs not addressed in the command request, these events are treated like autonomous events. For further details see corresponding section in the transmission chapter – entity cross-connection.

Example input/output

The following shows an example of the **SET-SID** command where the target/source identification of the NE was changed to HARRY:

```
SET-SID:LT-UNITE::123456::HARRY;

   LT-UNITE 00-11-15 08:00:00
M  123456 COMPLD
;
```

Error responses

Refer to the section “Error responses” of “RTRV-HDR: Retrieve header” (p. 3-2417). The error responses listed there also apply to the **SET-SID** command.

Validation rule: Check of Maintenance Condition

If a SET-SID command is entered when the administrative state of the system is not in maintenance condition, the following error response is returned.

Error code:

`SNVS – Status, Not in Valid State, not in Maintenance Condition`

Validation rule: Uniqueness of TID within Ringmaps

If the new sid conflicts with an existing TID in at least one of the valid ringmaps, the following error response is returned.

Error code:

`IDNV – Input, Data Not Valid, sid is already used by another BLSR/MS-SPRing Node`

Related TL1 messages

“RTRV-HDR: Retrieve header” (p. 3-2417)

References

GR-833-CORE, ″Surveillance Messages″

SET-SID notification

Output format

Syntax: **SET-SID**::*newsid*

Output parameters

Command parameters

1. **newsid** – New Target/Source Identification
   See definition of input parameter *tid* in “RTRV-HDR: Retrieve header” (p. 3-2417).

Example output

```
SET-SID:LT-UNITE::805::HARRY;

   LT-UNITE  00-11-17  10:11:00
M  805  COMPLD;


   LT-UNITE  00-11-17  10:12:03
A  610  REPT  DBCHG
   Time=13-49-30,DATE=00-11-17:SET-SID::HARRY”
;
```

SET-SYNCN: Set synchronization

The **SET-SYNCN** command is used to provision system timing clock mode, assign timing references, and external input and output timing port attributes. The command is used not only to assign system timing attributes but also to modify or remove them.

Properties

Login privilege: User Privilege Code (UCFC/UCAL): M3

Priority: 1

Abortable: No.

Input format

Command syntax: **SET-SYNCN**:*tid*:*aid*:*ctag*:::*spec_block*;

Input parameters

Command parameters

1. **tid** – target identifier
   See definition of *tid – target identifier* in "RTRV-HDR: Retrieve header" (p. 3-2417).
2. **aid** – access identifier
   This parameter specifies the timing related settings. The AID separates the parameter in system timing related parameter, parameters related to assigned timing references and parameters for port provisioning. The values all or *someprefix* all are not allowed in this context.
   Type: AID
   Range: SystemAID, External_timing_ref_AID, Line_timing_ref_AID, External_Timing_input_port_AID, External_Timing_output_port_AID
   Default: Not applicable.
3. **ctag** – correlation tag
   See definition of *ctag – correlation tag* in "RTRV-HDR: Retrieve header" (p. 3-2417).

Spec block parameters

1. System Timing
   a. **clkmod** – clock mode
      This parameter specifies the provisioned operational mode of the system timing.
      In the clockmod = FREE-RUNNING mode the system clock process uses the internal oscillator as reference. In the clockmod = LOCKED mode the system clock process uses the input signal from the system.
      Type: Alphanumeric value set.
      Range: FREE-RUNNING, LOCKED.

Default: FREE-RUNNING.

b. **syncmsg** – system synchronization status messaging mode

This parameter is used to ENABLE or DISABLE the SSM protocol.

The parameter provisions the automatic reference selection process to use the incoming QLs in QL-enabled mode or to ignore them in QL-disabled mode. In case of DISABLE, for SDH all output SSMs are set to DNU, for SONET all output SSMs are set to STU.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: ENABLE.

c. **timing** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

Default: Default

d. **wtr** – wait to restore

This parameter specifies the provisioned wait to restore timer used for all timing ports for revertive switching. This is the amount of time it takes to switch from lower priority to higher priority references. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: 0SEC, 20SEC, 0…60MIN (1 minute steps), 99 (infinite).

SDH: 0SEC, 0…60MIN (1 minute steps); the values “0SEC” and “0MIN” can be used for SONET and SDH to provision a zero wait to restore time.

Default:

SONET: 20SEC

SDH: 5MIN

2. Assigned Timing Reference

a. **extout_pri** – timing output reference priority

This parameter specifies the priority assignment of the timing link switch of the external output timing. This parameter is only for SDH and for line timing references.

The greater the number the lower the priority. The number ″1″ indicates the highest priority. The priority ″0″ means not assigned for external output.

Type: Numeric value set.

Range: 0 (means DISABLED), 1, 2, 3, 4, 5, 6.

Default: 0 (means DISABLED).

b. **ql_prov** – quality provisioned

This parameter defines the quality for the specified timing reference to a fixed SSM value. It is the provisioned quality level sync message value that will be received on timing references.

This parameter is related to all assigned timing references (both external timing references and line timing references) and overrides what is received for the respective timing reference. The AUTO value implies sync message support. The AUTO value is rejected if SSM is not supported for the assigned timing reference (i.e. for DS1 SF). The value range depends from the system synchronization characteristic. Per default is:

AUTO – for STM-N/OC-N line, 2Mbit/s framed or DS1 ESF with SSM,

SSUT – for 2MHz, 2Mbit/s unframed or framed without SSM

STU – for DS1 SF or DS1 ESF without SSM

Type: Alphanumeric value set.

Range:

SONET: AUTO, PRS, STU, ST2, ST3

SDH: AUTO, PRC, SSUL, SSUT, SEC.

Default:

SONET: AUTO (External_timing_ref_AID, Line_timing_ref_AID)

SDH: AUTO (Line_timing_ref_AID) SSUT (External_timing_ref_AID).

c. **`syncref_pri`** – system timing reference priority

   This parameter defines the priority for assigned timing references.

   When multiple input references have been given the same priority value, it implies non-revertive switching between the associated references. Otherwise, it implies revertive switching to the reference with the highest assigned priority. When the QL-control is disabled, the selection process is controlled by signal status and port priority. The greater the number, the lower the priority. Priority ″0″ means not assigned.

   Type: Numeric value set.

   Range: 0 (means DISABLE), 1, 2, 3, 4, 5, 6, 7, 8.

   Default: 0 (means DISABLE).

d. **`timing`** – ASAP name

   This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

   Type: Alphanumeric value set.

   Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

   Default: Default

e. **`tpmode`** – timing port mode monitoring

   This parameter specifies the monitoring status of the timing port. If the timing port is in the MONITORED mode and the reference is in FAILED status alarms are active for this port. The value NOT-MONITORED means that alarm monitoring is not occurring on the port.

   Type: Alphanumeric value set.

   Range: MONITORED, NOT-MONITORED.

Default: NOT-MONITORED.

f. **`tref_aid`** – timing reference assigned port AID

This parameter specifies the port AID of the assigned timing reference AID.

The parameter is rejected, if the timing reference already assigned to a timing port, or if the reference does not support SSM and timing port is provisioned AUTO, or if the unit which delivers the reference is not present or pre-provisioned. It is not possible to assign more then one reference from a unit. It is not possible to un-assign units from which a reference is assigned to a timing port. It is also not possible to assign a reference from a protection port in a protection group. The value NOT-CONNECTED means no reference is assigned to current timing port.

Type: AID.

Range: External_Timing_input_port_AID, SPortAID, NOT-CONNECTED.

Default: NOT-CONNECTED.

3. External Timing Input

a. **`ext_if`** – external input signal format

This parameter specifies the format of the external timing input.

The value range depends on the system synchronization characteristic. The external timing input format can be set at both external timing inputs individually. The reference is rejected, if the reference does not support SSM and the timing port is provisioned AUTO.

Type: Alphanumeric value set.

Range:

SONET: SF, ESF

SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

Default:

SONET: ESF

SDH: 2MHz.

b. **`ext_isa`** – external input sa bit location

This parameter defines the external input Sa bit location. It is only valid for the external timing input format “2Mbit/s” framed; i.e. for ports with format unequal to “2Mbit/s framed” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

The location of the Sa bit is can be set at both external timing inputs individually. One of the Sa4 to Sa8 bits of the 2Mbit/s sub-multiframe is allocated for Synchronization Status Messages (SSM).

Type: Alphanumeric value set.

Range: SA4, SA5, SA6, SA7, SA8.

Default: SA4.

4. External Timing Output

a. **`ext_equ`** – cable equalization

This parameter specifies the timing output cable equalization to adjust the specified timing reference output signal power. This parameter is valid only for DS1 (SONET).

This parameter is rejected for all input interfaces except DS1. The values are represented by percentage (3.0dB=100%; 2.4dB=80%; 1.8dB=60%; 1.2dB=40%; 0.6dB=20%).

Type: Numeric value set.

Range: 20, 40, 60, 80, 100.

Default: 20.

b. **`ext_of`** – external output signal format

This parameter specifies the external output signal format. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: SF, ESF

SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

Default:

SONET: ESF

SDH: 2MHz.

c. **`ext_osa`** – external output sa bit location

This parameter defines the external output Sa bit location. This parameter is only valid for external timing output format “2Mbit/s framed”; i.e. for ports with format unequal to “2Mbit/s framed” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.

Range: SA4, SA5, SA6, SA7, SA8.

Default: SA4.

d. **`extout_en`** – external timing output enabling/disabling

This parameter specifies whether the external timing output is enabled or disabled.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: DISABLE.

e. **`extout_prov`** – external timing output selection

This parameter specifies the operation mode of the external output timing.

The value SETG means the system clock. The value TLS means the output. For SONET both values are fixed to the respective external timing output. The value range depends on the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]

SDH: SETG, TLS.

Default: SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]

SDH: SETG.

f. **`extout_ssm`** – external timing output AIS mode

This parameter specifies which value will be send when the signal level is below the quality acceptance level or when *frcdus* is enabled. The parameter is valid depending from the system synchronization characteristic, for SDH 2Mbit/s framed and for SONET DS1 ESF; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

The value AISMODE means AIS is inserted in the external timing output signal. The value QLMODE means DUS is inserted in the external timing output signal.

Type: Alphanumeric value set.

Range: AISMODE, QLMODE.

Default: QLMODE.

g. **`frcdus`** – forced DNU/DUS

This parameter specifies the synchronization messaging forced DNU/DUS enable or disable for the timing output port. DUS applies to SONET and DNU applies to SDH. This is only valid for ’2Mbit/s framed’ and ’DS1 ESF’; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: DISABLE.

h. **`loopdelay`** – timing regenerator loop delay

This parameter indicates the delay in the SASE/BITS loop operation. The values are represented by seconds in steps of 1 seconds per loop.

Type: Numeric value set.

Range: 0…60 (1 seconds steps).

Default: 10.

i. **`qlout_tais`** – acceptance quality level for output threshold AIS

This parameter is the provisioned quality acceptance level for external timing output. When the incoming message is below this level, AIS or DUS (depending from the *extout_ssm* value) will be inserted into the external timing output. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: PRS, STU, ST2, ST3

SDH: PRC, SSUT, SSUL, SEC.

Default:

SONET: ST3

SDH: SEC.

j. **`regen_loop`** – timing regenerator loop

This parameter specifies the avoiding of loops over the SASE/BITS clocks.

When this parameter is set to ENABLED, the synchronization message DUS will be inserted in the traffic output port associated with the traffic input port used as source for the station clock. The parameter is selectable for both external timing outputs individually.

Type: Alphanumeric value set.

Range:

SONET: DISABLE, ENABLE

SDH: DISABLE, EXTREF1, EXTREF2, BOTH.

Default:

SONET: ENABLE

SDH: DISABLE.

k. **`timing`** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

Default: Default

Output format

After receiving the **`SET-SYNCN`** command, the following normal completion response is returned:

```
   sid date time
M  ctag COMPLD
;
```

The output parameters *sid*, *date*, *time*, and *ctag* included in the normal completion response are specified in the section “Output parameters” of “RTRV-HDR: Retrieve header” (p. 3-2417).

Detailed behavior description

After successful receiving the **`SET-SYNCN`** command a **`REPT DBCHG`** database change notification is returned. This notification consists the same parameter as per delivered by the command. Notifications are sent whenever a user-provisionable parameter has been changed.

When the system receives a SET-SYNCN command with changing of the clock mode, a REPT EVT report event notification is returned.

If the system receives a SET-SYNCN command with changing of the active timing reference, a REPT EVT report event notification is returned.

It is not possible to assign a reference from a protection port in a protection group.

If the system synchronization characteristic is SONET and the SSM is invalid the value DUS is displayed for QL values.

It is possible to provision a QL acceptance level to the external timing output. If the system quality below this acceptance QL level and QL control is enabled and the external timing output is enabled then an AIS signal or DUS, depending on settings of parameter *extout_ssm*, is substituted in the output.

If the system timing QL mode is DISABLED for SDH all output SSMs are set to DNU, for SONET all output SSMs are set to STU.

If the system synchronization characteristic is SDH and the external timing input format is either 2MHz or 2Mbit/s unframed or 2Mbit/s framed without SSM then the assigned timing reference status are default SSUT. If the system synchronization characteristic is SONET and the external timing input format is either SF or ESF without SSM then the assigned timing reference status are default STU. The value AUTO is rejected if SSM is not supported for the assigned timing reference.

It is possible to provision the automatic reference selection process to use the incoming QLs in QL-enabled mode. The input with the highest quality level is selected. If there is more then one input with the same highest quality level, then the input with the highest port priority is selected. In QL-disabled mode the reference selection is working by priority provisioning. It is possible to assign the same priority to multiple inputs. In this case, it implies non revertive switching between the associated references.

A MS-SPRing/BLSR span or MSP pair is handled as one transport port, so also counts as one reference signal. If one of the sections of the pair do not support SSM, the reported QL status is NOT-SUPPORTED.

If the timing port is in the NOT-MONITORED mode then no Timing Link Fail alarm will be active when reference is in FAILED status or no Timing Link Unequipped alarm will be active when the reference is in NOT-CONNECTED state.

If the external input signal format ext_if is changed to an unframed mode without also specifying the external input ssm support ext_issm then the external input ssm support ext_issm is set to NOT-SUPPORTED autonomously if not already having this value. The change is reported via a SET-SYNCN notification.

Example input/output

The following is an example of the **SET-SYNCN** command for System Timing:

```
SET-SYNCN:LT-UNITE:system:123456:::clkmod=LOCKED,wtr=1MIN;

   LT-UNITE 00-11-14 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **SET-SYNCN** command for Assigned Timing Reference:

```
SET-SYNCN:LT-UNITE:extref1:123456:::tref_aid=EXTTMG0,
ql_prov=SSUL;

   LT-UNITE 00-11-14 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **SET-SYNCN** command for External Timing Input:

```
SET-SYNCN:LT-UNITE:exttmg1:123456:::ext_if=2MHZ;

   LT-UNITE 00-11-14 12:00:00
M  123456 COMPLD
;
```

The following is an example of the **SET-SYNCN** command for External Timing Output:

```
SET-SYNCN:LT-UNITE:exttmg0_out:123456:::ext_of=2MHZ,extout_en=ENABLE;

   LT-UNITE 00-11-14 12:00:00
M  123456 COMPLD
;
```

Error responses

Validation rule: Timing reference already assigned

If a timing reference already assigned to a timing port or circuit pack the system responds with:

Error code: `SDNC – Status, Data Not Consistent, reference already assigned`

Validation rule: One timing port for a circuit pack

If a second port of a circuit pack is assigned to a timing reference the system responds with:

Error code:

`SDNC – Status, Data Not Consistent, other port already assigned`

Validation rule: No timing port on protection pack of an MS protected timing port

If another port of a circuit pack where the protection port of an MS protection group is located whose associated worker port for this span has already a timing reference being assigned the system shall respond with:

Error code:

`SDNC – Status, Data Not Consistent, other port already assigned`

″MS protection group″ denotes an MSP/APS or a 4 fibre MS SPRING/BLSR group.

Validation rule: No out-of-service state for timing port

If an ’out of service’ port (i.e. the port attribute *servcond* has the value ’OUTOFSERVICE’) is assigned to a timing reference the system responds with:

Error code: `SDNC – Status, Data Not Consistent, port is out-of-service`

A port must be in the out-of-service before any port loop can be switched. Therefore it is sufficient to request for a timing reference assignment, that a port is not in the out-of-service state in order to prevent timing loops caused by port loops; i.e. a port that is not in out-of-service state will not have a switched port loop.

Validation rule: Timing reference from a protection port in a protection group only allowed in asymmetric spans of a ring

If a timing reference is assigned from a protection port in a protection group which is not part of an asymmetric span, the system shall respond with:

Error code:
`SDNC – Status, Data Not Consistent, reference in protection group`

The characteristic of an asymmetric ring is that one or more spans have only protection lines, no working lines (asymmetric spans). In these asymmetric spans it must be possible to use the protection ports for timing.

Validation rule: Port is already assigned to another timing reference

Ports can be assigned only one. If a port already assigned to a timing reference the system responds with:

Error code: `SDNC – Status, Data Not Consistent, port already assigned`

Validation rule: Command rejection if quality evaluation disabled

If the quality mode disabled the setting to forced DUS/DNU or the setting the quality output threshold AIS is rejected and the system responds with:

Error code:
`SDNC – Status, Data Not Consistent, quality evaluation disabled`

Validation rule: No transparent port as line timing reference

The system shall not allow to use a Transparent Port, i.e. ports with a rate of STM16T or OC48T, as line timing reference.

Error code:
`IDNV – Input, Data Not Valid, Transparent Ports not allowed as line timing reference`

Validation rule: No DS3 Port as line timing reference

The system shall not allow to use a DS3 Port, i.e. a port with a rate of T3, as line timing reference.

Error code:
`IDNV – Input, Data Not Valid, DS3 Ports not allowed as line timing reference`

Validation rule: No EC1 Port as line timing reference

The system shall not allow to use a EC1 Port as line timing reference.

Error code:
`IDNV – Input, Data Not Valid, EC1 Ports not allowed as line timing reference`

Validation rule: SSM not supported

SSM support is rejected for invalid input or output format types and the system responds with:

Error code:
`IDNV – Input, Data Not Valid, SSM not supported for this format`

Validation rule: SSM not supported for unframed formats

Setting an invalid combination of external input frame format (ext_if) and external input SSM support (ext_issm) by changing one or both parameters are rejected and the system shall respond with:

Error code:
`IDNV – Input, Data Not Valid, SSM not supported for this format`

Validation rule: Format not supported for forced DNU/DUS

DNU/DUS provisioning is rejected for invalid output format types and the system responds with:

Error code:
`IDNV – Input, Data Not Valid, forced DUS not supported for this format`

Validation rule: Input / Output format not supported for Sa bit settings

Sa bit location setting is rejected for invalid input or output format types and the system responds with:

Error code: `IDNV – Input, Data Not Valid, Sa bit location not supported`

Validation rule: SSM not supported if timing port provisioned AUTO

If a timing port provisioned AUTO and the reference which is being assigned does not support SSM, the assignment request is rejected and the system responds with:

Error code:
`IDNV – Input, Data Not Valid, SSM not supported for provisioned QL`

Validation rule: Output type not supported for AIS mode

If the AIS mode provisioned to a not supported output type then the system responds with:

Error code: `IDNV – Input, Data Not Valid, AIS mode not valid`

Validation rule: Cable equalization not supported

If the cable equalization provisioned to a not supported output type or if the system synchronization characteristic is SDH then the system responds with:

Error code: `IDNV – Input, Data Not Valid, cable equalization not supported`

Validation rule: Timing clock mode inconsistent

If the NE receives a command with FREE-RUNNING provisioning while the clock mode switch request state is forced holdover, the following error response is returned:

Error code:
`SNVS – Status, Not in Valid State, timing clock mode inconsistent`

Validation rule: Regenerator loops with SETG as timing output

If the output timing source selection is set to SETG and the timing regenerator loop is provisioned, then the system responds with:

Error code:
`IDNV – Input, Data Not Valid, no regen loop with SETG as timing output`

Related TL1 messages

- "RTRV-SYNCN: Retrieve synchronization" (p. 3-2955)
- "OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
- "RLS-SYNCNSW: Release synchronization switch" (p. 3-2215)

References

GR-833-CORE

SET-SYNCN notification

Output format

The **SET-SYNCN** notification has the following specific syntax:

SET-SYNCN:*aid*::*spec_block*

Output parameters

Command parameters

1. **aid** – access identifier
   The AID is used to specify the timing related settings. The AID separates the parameter in system timing related parameter, parameters related to assigned timing references and parameters for port provisioning. The values all or someprefix all are not allowed in this context.
   Type: AID
   Range:
   - SystemAID
   - External_timing_ref_AID
   - Line_timing_ref_AID
   - External_Timing_input_port_AID
   - External_Timing_output_port_AID

Spec block parameters

1. System Timing
   a. **clkmod** – clock mode
      This parameter specifies the provisioned operational mode of the system timing. In the *clockmod* = FREE-RUNNING mode the system clock process uses the internal oscillator as reference. In the *clockmod* = LOCKED mode the system clock process uses the input signal from the system.
      Type: Alphanumeric value set.
      Range: FREE-RUNNING, LOCKED.
      Default: FREE-RUNNING.
   b. **syncmsg** – system synchronization status messaging mode
      This parameter is used to ENABLE or DISABLE the SSM protocol.
      The parameter provisions the automatic reference selection process to use the incoming QLs in QL-enabled mode or to ignore them in QL-disabled mode. In case of DISABLE, for SDH all output SSMs are set to DNU, for SONET all output SSMs are set to STU.
      Type: Alphanumeric value set.
      Range: ENABLE, DISABLE.
      Default: ENABLE.
   c. **timing** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

Default: Default

d. **`wtr`** – wait to restore

This parameter specifies the provisioned wait to restore timer used for all timing ports for revertive switching. This is the amount of time it takes to switch from lower priority to higher priority references. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: 0SEC, 20SEC, 0…60MIN (1 minute steps), 99 (infinite).

SDH: 0SEC, 0…60MIN (1 minute steps); the values “0SEC” and “0MIN” can be used for SONET and SDH to provision a zero wait to restore time.

Default:

SONET: 20SEC

SDH: 5MIN

2. Assigned Timing Reference

a. **`extout_pri`** – timing output reference priority

This parameter specifies the priority assignment of the timing link switch of the external output timing. This parameter is only for SDH and for line timing references.

The greater the number the lower the priority. The number ″1″ indicates the highest priority. The priority ″0″ means not assigned for external output.

Type: Numeric value set.

Range: 0 (means DISABLED), 1, 2, 3, 4, 5, 6.

Default: 0 (means DISABLED).

b. **`ql_prov`** – quality provisioned

This parameter defines the quality for the specified timing reference to a fixed SSM value. It is the provisioned quality level sync message value that will be received on timing references.

This parameter is related to all assigned timing references (both external timing references and line timing references) and overrides what is received for the respective timing reference. The AUTO value implies sync message support. The AUTO value is rejected if SSM is not supported for the assigned timing reference (i.e. for DS1 SF). The value range depends from the system synchronization characteristic. Per default is:

AUTO – for STM-N/OC-N line, 2Mbit/s framed or DS1 ESF with SSM,

SSUT – for 2MHz, 2Mbit/s unframed or framed without SSM

STU – for DS1 SF or DS1 ESF without SSM

Type: Alphanumeric value set.

Range:

SONET: AUTO, PRS, STU, ST2, ST3

SDH: AUTO, PRC, SSUL, SSUT, SEC.

Default:

SONET: AUTO (External_timing_ref_AID, Line_timing_ref_AID)

SDH: AUTO (Line_timing_ref_AID) SSUT (External_timing_ref_AID).

c. **`syncref_pri`** – system timing reference priority

This parameter defines the priority for assigned timing references.

When multiple input references have been given the same priority value, it implies non-revertive switching between the associated references. Otherwise, it implies revertive switching to the reference with the highest assigned priority. When the QL-control is disabled, the selection process is controlled by signal status and port priority. The greater the number, the lower the priority. Priority ″0″ means not assigned.

Type: Numeric value set.

Range: 0 (means DISABLE), 1, 2, 3, 4, 5, 6, 7, 8.

Default: 0 (means DISABLE).

d. **`timing`** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.

Default: Default

e. **`tpmode`** – timing port mode monitoring

This parameter specifies the monitoring status of the timing port. If the timing port is in the MONITORED mode and the reference is in FAILED status alarms are active for this port. The value NOT-MONITORED means that alarm monitoring is not occurring on the port.

Type: Alphanumeric value set.

Range: MONITORED, NOT-MONITORED.

Default: NOT-MONITORED.

f. **`tref_aid`** – timing reference assigned port AID

This parameter specifies the port AID of the assigned timing reference AID.

The parameter is rejected, if the timing reference already assigned to a timing port, or if the reference does not support SSM and timing port is provisioned AUTO, or if the unit which delivers the reference is not present or pre-provisioned. It is not possible to assign more then one reference from a unit. It is not possible to un-assign units from which a reference is assigned to a timing port. It is also not possible to assign a reference from a protection port in a protection group. The value NOT-CONNECTED means no reference is assigned to current timing port.

Type: AID.

Range: External_Timing_input_port_AID, SPortAID, NOT-CONNECTED.

Default: NOT-CONNECTED.

3. External Timing Input
   a. **`ext_if`** – external input signal format

      This parameter specifies the format of the external timing input.

      The value range depends on the system synchronization characteristic. The external timing input format can be set at both external timing inputs individually. The reference is rejected, if the reference does not support SSM and the timing port is provisioned AUTO.

      Type: Alphanumeric value set.

      Range:

      SONET: SF, ESF

      SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

      Default:

      SONET: ESF

      SDH: 2MHz.
   b. **`ext_isa`** – external input sa bit location

      This parameter defines the external input Sa bit location. It is only valid for the external timing input format "2Mbit/s" framed; i.e. for ports with format unequal to "2Mbit/s framed" the parameter value is retrievable, but it has no impact on the ports transmission behavior.

      The location of the Sa bit is can be set at both external timing inputs individually. One of the Sa4 to Sa8 bits of the 2Mbit/s sub-multiframe is allocated for Synchronization Status Messages (SSM).

      Type: Alphanumeric value set.

      Range: SA4, SA5, SA6, SA7, SA8.

      Default: SA4.
4. External Timing Output
   a. **`ext_equ`** – cable equalization

      This parameter specifies the timing output cable equalization to adjust the specified timing reference output signal power. This parameter is valid only for DS1 (SONET).

      This parameter is rejected for all input interfaces except DS1. The values are represented by percentage (3.0dB=100%; 2.4dB=80%; 1.8dB=60%; 1.2dB=40%; 0.6dB=20%).

      Type: Numeric value set.

      Range: 20, 40, 60, 80, 100.

      Default: 20.
   b. **`ext_of`** – external output signal format

      This parameter specifies the external output signal format. The value range depends from the system synchronization characteristic.

      Type: Alphanumeric value set.

      Range:

      SONET: SF, ESF

      SDH: 2MHz, 2Mbit-unframed, 2Mbit-framed.

Default:

SONET: ESF

SDH: 2MHz.

c. **`ext_osa`** – external output sa bit location

This parameter defines the external output Sa bit location. This parameter is only valid for external timing output format “2Mbit/s framed”; i.e. for ports with format unequal to “2Mbit/s framed” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.

Range: SA4, SA5, SA6, SA7, SA8.

Default: SA4.

d. **`extout_en`** – external timing output enabling/disabling

This parameter specifies whether the external timing output is enabled or disabled.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: DISABLE.

e. **`extout_prov`** – external timing output selection

This parameter specifies the operation mode of the external output timing.

The value SETG means the system clock. The value TLS means the output. For SONET both values are fixed to the respective external timing output. The value range depends on the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]

SDH: SETG, TLS.

Default: SONET: LINE1 [fix for output 1], LINE2 [fix for output 2]

SDH: SETG.

f. **`extout_ssm`** – external timing output AIS mode

This parameter specifies which value will be send when the signal level is below the quality acceptance level or when *frcdus* is enabled. The parameter is valid depending from the system synchronization characteristic, for SDH 2Mbit/s framed and for SONET DS1 ESF; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

The value AISMODE means AIS is inserted in the external timing output signal. The value QLMODE means DUS is inserted in the external timing output signal.

Type: Alphanumeric value set.

Range: AISMODE, QLMODE.

Default: QLMODE.

g. **`frcdus`** – forced DNU/DUS

This parameter specifies the synchronization messaging forced DNU/DUS enable or disable for the timing output port. DUS applies to SONET and DNU applies to SDH. This is only valid for ’2Mbit/s framed’ and ’DS1 ESF’; i.e. for ports with format unequal to “2Mbit/s framed” or “DS1 ESF” the parameter value is retrievable, but it has no impact on the ports transmission behavior.

Type: Alphanumeric value set.

Range: ENABLE, DISABLE.

Default: DISABLE.

h. **`loopdelay`** – timing regenerator loop delay

This parameter indicates the delay in the SASE/BITS loop operation. The values are represented by seconds in steps of 1 seconds per loop.

Type: Numeric value set.

Range: 0…60 (1 seconds steps).

Default: 10.

i. **`qlout_tais`** – acceptance quality level for output threshold AIS

This parameter is the provisioned quality acceptance level for external timing output. When the incoming message is below this level, AIS or DUS (depending from the *extout_ssm* value) will be inserted into the external timing output. The value range depends from the system synchronization characteristic.

Type: Alphanumeric value set.

Range:

SONET: PRS, STU, ST2, ST3

SDH: PRC, SSUT, SSUL, SEC.

Default:

SONET: ST3

SDH: SEC.

j. **`regen_loop`** – timing regenerator loop

This parameter specifies the avoiding of loops over the SASE/BITS clocks.

When this parameter is set to ENABLED, the synchronization message DUS will be inserted in the traffic output port associated with the traffic input port used as source for the station clock. The parameter is selectable for both external timing outputs individually.

Type: Alphanumeric value set.

Range:

SONET: DISABLE, ENABLE

SDH: DISABLE, EXTREF1, EXTREF2, BOTH.

Default:

SONET: ENABLE

SDH: DISABLE.

k. **`timing`** – ASAP name

This parameter specifies the name of the ASAP (Alarm Severity Assignment Profile) that is assigned to timing.

Type: Alphanumeric value set.

Range: PFNAME (alphanumeric string with 1… 24 characters). A hyphen (“-”) is allowed in the name additionally.
Default: Default

Example output

The following is an example of the SET-SYNCN notification for System Timing:

```
   LT-UNITE 00-11-16 12:00:00
A  000 REPT DBCHG
   ”TIME=12-00-00,DATE=00-11-16:SET-SYNCN:SYSTEM::wtr=2MIN”
;
```

The following is an example of the SET-SYNCN notification for Assigned Timing Reference:

```
   LT-UNITE 00-11-16 12:00:00
A  000 REPT DBCHG
   ”TIME=12-00-00,DATE=00-11-16:SET-SYNCN:LINE2::ql_prov=PRC”
;
```

The following is an example of the SET-SYNCN notification for External Timing Input:

```
   LT-UNITE 00-11-16 12:00:00
A  000 REPT DBCHG
   ”TIME=12-00-00,DATE=00-11-16:SET-SYNCN:EXTTMG0::ext_isa=SA5,
ext_if=2MBIT-FRAMED”
;
```

The following is an example of the SET-SYNCN notification for External Timing Output:

```
   LT-UNITE 00-11-16 12:00:00
A  000 REPT DBCHG
   ”TIME=12-00-00,DATE=00-11-16:SET-SYNCN:EXTTMG1_OUT::ql_out=SEC,
extout_ssm=AISMODE”
;
```

TCA notification

Output format

Command syntax: **`REPT EVT`** *modifier aid:condtype,condeff,ocrdat,ocrtm,,,,thlev,tmper:conddescr*

Output parameter

1. **`modifier`**
   This parameter indicates the port type or tributary on which the **`INIT-REG`** command is to act on.
   Type: Alphanumeric value set.
   Range:
   The following values are for SONET only:
   - STS1, STS3C, STS12C, STS48C, STS192C, OC3, OC12, OC48, OC48T, OC192, OC768, T3, EC1, LOVT1.

   The following values are for TransMUX only:
   - TMAT3, TMASTS1, DS1

   The following values are for SDH only:
   - VC3, VC4, VC44C, VC416C, VC464C, STM1E, STM1, STM4, STM16, STM16T, STM64, STM256, ODU1, LOVC3, LOVC12

   The following value is for Ethernet only:ENET, 1GEPL, 10GEPL.
2. **`aid`** – Access Identifier
   Type: AID
   Range:
   - For STM<N>, STM<N>T, OC<N>, EC1 and OC<N>T ports: SPortAID
   - For Ethernet ports: EPortAID VCGAID
   - For DS3 ports: APortAID, TMuxAPortAID
   - For DS1 NIM: DS1NumAID
   - For ODU terminations: VCGAID
   - For STS/VC tribs: STS1NumAID, TMuxAPTNumAID
   - For VCG tribs: VCGNumAID
   - For LOXC related LO tribs: GenLOTribAID.
3. **`condeff`** – Condition Effect
   This parameter indicates the effect of the reported event on the NE. It is used to distinguish between persistent and transient alarms/events.

Two state transitions can be distinguished for a persistent alarm/event:

- raise transition: state change from not present to present
- clear transition: state change from present to not present.

A transient alarm/event has no 'presence' state; i.e. just the occurrence of that event/alarm is reported. ABN condition are always persistent and not transient.
Type: Alphanumeric value set.

Range:

- CL: Standing condition cleared for a persistent alarm/event.
- SC: Standing condition raised for a persistent alarm/event
- TC: Transient condition occurred for a transient alarms/events.

4. **`conddescr`** – Description Of The Condition
   The general format is: category, entity, description, additional text. If there is no additional text, then the trailing comma may be omitted.
   Type: inner string

   Range:

   - For STM<N>, STM<N>T, OC<N>T, OC<N> and EC1: Quality of service, {rr} port, TCA where rr is equal to the port rate
   - For ODU<N>: Quality of service, {rr} path, TCA where rr is equal to the ODU rate (e. g., ODU1)
   - For not terminated VC12, VC3, VC4, VC44C, VC416C, VC464C, VT1, STS1, STS3C, STS12C, STS48C and STS192C: Quality of service, {rr} CS, TCA where rr is equal to the signal rate
   - For terminated VC4 and STS1signals: Quality of service, {rr} path, TCA where rr is equal to the Constituent Signal (CS) rate
   - For GE port: Quality of service, {ge_rate}GE, TCA where ge_rate is the signal rate of the GE port
   - For VCG port: Quality of service, VCG, TCA.
   - For DS3 port: Quality of service, T3 port, TCA
   - For TMAT3 port: Quality of service, T3 port, TCA
   - For TMASTS1 path termination: Quality of service, {***`rr`***} path, TCA, where ***`rr`*** is equal to STS1 rate
   - For not terminated DS1 tributary: Quality of service, DS1 path, TCA

5. **`condtype`** – Condition Type
   This parameter identifies the type of event indication being reported.
   Type: inner string with the Format: T-<montype>

   Range:

   - Range for SDH signals

     The following values are for all *STM<N> signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-MS-N-ES, T-MS-N-SES, T-MS-N-BBE, T-MS-N-UAS, T-MS-F-ES, T-MS-F-SES, T-MS-F-BBE, T-MS-F-UAS

The following values are for all *STM<N>T signals*: T-RS-N-ES, T-RS-N-SES, T-RS-N-BBE, T-RS-N-UAS, T-RS-N-ES-EGR, T-RS-N-SES-EGR, T-RS-N-BBE-EGR, T-RS-N-UAS-EGR

The following values also are for *STM64 and STM256 signals*: T-MS-N-FECC

The following values are for *VC3, VC4, VC44C, VC416C and VC464C tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, cVC-F-BBE, T-VC-F-UAS, T-PJCPOS, T-PJCNEG

The following values are for *VC12 tributaries*: T-VC-N-ES, T-VC-N-SES, T-VC-N-BBE, T-VC-N-UAS, T-VC-F-ES, T-VC-F-SES, T-VC-F-BBE, T-VC-F-UAS

- Range for SONET signals

  The following values are for all *OC<N> and EC1 signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVL, T-ESL, T-SESL, T-UASL, T-CV-LFE, T-ES-LFE, T-SES-LFE, T-UAS-LFE

  The following values are for all *OC<N>T signals*: T-CVS, T-ESS, T-SESS, T-SEFS-S, T-CVS-EGR, T-ESS-EGR, T-SESS-EGR, T-SEFS-S-EGR

  The following values are also for *OC192 and OC768 signals*: T-FECC-L

  The following values are for STS1, STS3C, STS12C, STS48C and STS192C tributaries: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE, PPJC-PDet, NPJC-PDet, NPJC-PDet, NPJC-PGen

  The following values are for *TMASTS1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP, T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

  The following values are for *VT1 tributaries*: T-CVP, T-ESP, T-SESP, T-UASP,T-CV-PFE, T-ES-PFE, T-SES-PFE, T-UAS-PFE

- Range for DS3/DS1 signals

  The following values are for *DS3 signals with T3 modifier*:

  T-CVL, T-ESL, T-ESA-L, T-ESB-L, T-SESL, T-CVP, T-ESP, T-ESA-P, T-ESB-P, T-SESP, T-UASP, T-SAS-P, T-CV-PFE, T-ES-PFE, T-ESA-PFE, T-ESB-PFE, T-SES-PFE, T-UAS-PFE, T-SAS-PFE, T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS3 signals with TMAT3 modifier*:

  T-CVP-EGR, T-ESP-EGR, T-ESA-P-EGR, T-ESB-P-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR, T-CV-PFE-EGR, T-ES-PFE-EGR, T-ESA-PFE-EGR, T-ESB-PFE-EGR, T-SES-PFE-EGR, T-UAS-PFE-EGR, T-SAS-PFE-EGR

  The following values are for *DS1 signals*:

  T-CVP-EGR, T-ESP-EGR, T-SESP-EGR, T-UASP-EGR, T-SAS-P-EGR

- Range for other signals

  The following values are for all *ODU<N> signals*:

  T-ODU-N-ES, T-ODU-N-SES, T-ODU-N-BBE, T-ODU-N-UAS, T-ODU-F-ES, T-ODU-F-SES, T-ODU-F-BBE, T-ODU-F-UAS

  For *ENET signals*:

  T-PDE, T-EDFE, T-EDFC

6. **ocrdat** – Occurrence Date
   This parameter indicates the raising / clearing date of the alarm.
   Type: value set of the format MM-DD.
   Range: MM represents the month, DD represents the day, leading zeros are not suppressed.
7. **ocrtm** – Occurrence Time
   This parameter indicates the raising / clearing time of the alarm. The accuracy should be +/- 1 second.
   Type: value set of the format HH-MM-SS.
   Range: HH represents the hours, MM represents the minutes, SS represents the seconds, leading zeros are not suppressed.
8. **thlev** – Threshold Level
   This parameter is only used for Threshold Crossing Alerts (TCAs). It reports the threshold level value which is set by ED-TCA-PROF.

   The two applicable thresholding methods (tcamd) must be distinguished. For TR-only mode:

   - the raise TCA threshold value is reported.

   For TR-RTR mode:

   - the raise TCA threshold value is reported for the raise event
   - the reset TCA threshold value is reported for the clear event.

   Type: Integer.
   Range: >1 (1 … 4294967295). Leading zeros will be suppressed.
9. **tmper** – Time Period
   This parameter reports the monitored time interval. This parameter is only used for Threshold Crossing Alerts (TCAs).
   Type: value set.
   Range: 15-MIN, 1-DAY.

Example input/output

TCA alarm for an STS-1 SES of slot 14 for port 3 STS-12 (24h register):

```
   UNITE-NE 00-01-01 08:00:50
A  009 REPT EVT STS1
   ″1-1-#-#-14-3-2:T-SESP,TC,01-10,08-00-49,,,,250,1-DAY:\″Quality of service, STS1
      CS, TCA\″″
;
```

4 TL1 messages - administration and security

Overview

Purpose

This section lists the TL1 interface definition for the domain “administration and security”.

Contents

Session

Overview

Purpose

This entity list the parameters which handle the establishment and removal of sessions between the NE and the Managers.

Contents

List of commands

NE security administration

Overview

Purpose

This entity lists the NE security parameters which are defined for the whole NE.

Contents

List of commands

- "RTRV-NE-SECU: Retrieve NE security" (p. 3-2552)
- "RTRV-BANNER" (p. 3-2272)
- "ED-NE-SECU: Edit NE security" (p. 3-446)
- "ED-NE-SECU notification" (p. 3-449)
- "ENT-BANNER" (p. 3-1452)
- "ENT-BANNER notification" (p. 3-1454)

User administration

Overview

Purpose

This entity refers to the User Security Data.

Contents

List of commands

Security Activity Log

Overview

Purpose

This entity lists to the security activity log data.

Contents

List of commands

5 TL1 messages - configuration management

Overview

Purpose

This chapter lists the TL1 interface definition for the domain “configuration management”.

Contents

System

Overview

Purpose

This section lists the system level configuration messages.

Contents

List of commands

Autonomous notifications

Change acceptable type list for CTL

During system startup it shall be detected which CTL type is plugged in.

Appropriate to this CTL the acceptable type list of the second controller slot shall be changed

- If the first controller is a CTL, the second controller can be CTL or CTL/2
- If the first controller is a CTL/2, the second controller must be CTL/2.

The term *first* and *second* is related on installation time. No database change notification will be send out during system initialization.

Change of the data base change status

The data base change state (see *dbstate* parameter in "RTRV-STATE-EQPT (system): Retrieve equipment state (system)" (p. 3-2845)) can change autonomously. This change is reported by means of an ED-STATE-EQPT notification.

System start-up with default DB

The system autonomously create the entity *system* when starting up with the default database. No notification is sent.

Bay

Overview

Purpose

This section lists the bay level configuration messages.

Contents

RTRV-EQPT: Retrieve equipment (bay)

For description of this parameter refer to "RTRV-EQPT (bay): Retrieve equipment (bay)" (p. 3-2373).

Autonomous notifications

System start-up with default DB

The system autonomously create the entity *bay* when starting up with the default database. No notification is sent.

Shelf

Overview

Purpose

This section lists the shelf level configuration messages.

Contents

List of commands

Autonomous notifications

System start-up with default DB

The system autonomously create the entity *shelf* when starting up with the default database. No notification is sent.

Slot

Overview

Purpose

This section lists the slot level configuration messages.

Contents

List of commands

Autonomous notifications

Circuit pack plug in

Autoprovisioning

If a circuit pack is plugged into an allowed slot, which is not provisioned to any circuit pack, this slot is auto-provisioned to this plugged-in pack. This change is reported by means of an ED-EQPT notification for this slot for the *nvmname, nvmqual* parameters and an ED-STATE-EQPT notification for the *holdst* parameter.

Unexpected, not acceptable, not recognizable CP handling

If an unexpected or not acceptable circuit pack or a circuit pack which inventory data can not be read is plugged into the slot only the holder status is changed and this is notified by an ED-STATE-EQPT notification.

For all these cases alarms will be sent out in addition.

Report of inventory data

In addition to the notifications above the system reports an ED-EQPT (slot) notification for the parameters *name, qual, app, ssn, clei, eci, sln*.

In case the inventory data can not be read the fields shall be reported as empty ("").

If circuit packs will be plugged in containing plugged in modules or fixed mounted modules implicit the socket entities will be updated and created according description of auto provisioning of modules.

Create CTL protection group- when plugging a second CTL

When plugging in a second CTL the system shall create one instance of 1+1 equipment protection group of type EQPTCTL

This shall be reported by means of ENT-PROTNGRP notification for 1+1 equipment protection group. The initial protection state shall be reported by means of a protection switch notification for the 1+1 equipment protection group.

Circuit pack plug out

If a circuit pack is plugged out the holder state for this board will change. This change is reported by means of an ED-STATE-EQPT notification for the *holdst* parameter.

In addition to the notification above the system reports an ED-EQPT (slot) notification for the parameters *name, qual, app, ssn, clei, eci, sln*.

If a circuit pack is plugged out which contains sockets with plugged in modules, then implicitly the socket entities shall be updated as required for the case of plug out of modules.

Change of functional type for neighbor slots

If a multislot circuit pack with *nvmname*=<XXX> is provisioned to slot n the parameters *nvmname*, *nvmqual* of each slots m to n-1 (n>m) change to <XXX>, EXTD and the parameter *actlname* is set to empty (i.e. the output contains one row with *actlname*="", see also Example Input/output section of ED-EQPT(slot) notification). This change is reported by means of ED-EQPT notification for slot m to n-1 for *nvmname*, *nvmqual*, *actlname* parameters.

If a circuit pack is provisioned and as side effect it is no longer possible to use one of the neighbor slot for a multislot CP, the parameter *actname* for this slot is changed and all multislot CPs are removed. This change is reported by means of an ED-EQPT notification.

If a multislot circuit pack of slot n is deprovisioned the parameters *nvmname*, *nvmqual* of each slots m to n-1 (n>m) change to UNDEFINED, UNDEF and the parameters *actlname* changes to list of acceptable CPs. This change is reported by means of ED-EQPT notifications for slot m to n-1 for *nvmname*, *nvmqual*, *actlname* parameters.

If a circuit pack is deprovisioned and as side effect it becomes now possible to use one of the neighbor slot for a multislot CP, the parameter *actname* for this slot is changed and all multislot CPs are added. This change is reported by means of an ED-EQPT notification.

System start-up with default DB

The system shall autonomously create all entities addressable by SlotAID when starting up with the default database. No notification is sent.

Socket

Overview

Purpose

Circuit packs can contain sockets where modules can be plugged in. For unite in the current version Pluggable Interfaces mainly optical modules will be plugged in. A module with a pluggable interface will be only related to one port (e.g. a STM64 port). The relation is done in that way that a socket number n is related to a port number n of the same slot (e.g.port 1-1-#-#-1-sc3 is related to socket 1-1-#-#-1-3) The port functionality (and the related e.g. tributaries) is not available as long no module is provisioned by a provisioning command or by plugging in. This means the related entities will be only available if a module is provisioned. A socket behaves similar like a slot. That means there can be inventory data of the module read out, it can be (pre- or auto-) provisioned and also the acceptable type list for the socket is available. Also the rules how and when a module can be provisioned or de provisioned are similar.

Contents

List of commands

Autonomous notifications

Module plug in

Auto provisioning

If a module will be plugged into a not provisioned socket the socket will be auto provisioned to the plugged in module if the plugged in module type matches the acceptable type list of the socked. This shall be reported by means of the ED-EQPT (socket) notification. In addition also all entities will be created and updated as described in the detailed behaviour of the ENT-EQPT (socket) command. The update of the holder status of the socket will be reported by means of the ED-STATE-EQPT (socket) notification.

Unexpected, not acceptable, not recognizeable module handling

If an unexpected or not acceptable module or a module from which inventory data can not be read is plugged into the socket only the holder status is changed and this is notified by an ED-STATE-EQPT(socket) notification. For all these cases alarms will be sent out in addition , either Optical Module Type Mismatch or Optical Module EPROM Access Fail.

Report of inventory data

In addition to the notifications above the system shall report an ED-EQPT (socket) notification for the affected parameters.

Affected parameters for Optical Modules with SFP or XFP EEPROM format:

- *app*,*clei*, *ccode*, *compatibilitybyte*, *connectortype*, *datecde*, *interchangemarker*, *name*, *physicalid*, *prodid*, *qual*, *transceivercode*, *uniquenumber*, *vendorname*, *vendoroui*, *vendorpartno*, *vendorrevlvl*, *vendorsln*, *wesserialno*,*wesvendorid*.

Affected parameters for Optical Modules with Small EEPROM format:

- *app*, *clei*, *eci*, *name*, *qual*, *sln*, *ssn*.

In case the inventory data can not be read the fields shall be reported as empty (“”).

Auto provisioning of CPs with plugged in modules

If a circuit pack is auto provisioned which contains already plugged in modules or fixed installed modules or an already provisioned circuit pack will be plugged in where the modules are not provisioned but already plugged in or fixed mounted then is the behaviour the same as the module will be auto provisioned by plugging in a module into a socket of the circuit pack. It can be seen as two independent actions, which will be performed one after the other: (auto) provisioning of the circuit pack with following auto provisioning of modules.

Module plug out

If a module is plugged out the holder state for this socket will change. This change shall be reported by means of an ED-STATE-EQPT (socket) notification for the holdst parameter.

In addition to the notification above the system reports an ED-EQPT (socket) notification for the affected parameters with the defined default value (empty string ″″).

Affected parameters for Optical Modules with SFP EEPROM format:

- *app,clei*, *ccode*, *compatibilitybyte*, *connectortype*, *datecde*, *interchangemarker*, *name*, *physicalid*, *prodid*, *qual*, *transceivercode*, *uniquenumber*, *vendorname*, *vendoroui*, *vendorpartno*, *vendorrevlvl*, *vendorsln*, *wesserialno*,*wesvendorid*.

Affected parameters for Optical Modules with Small or XFP EEPROM format:

- *app*, *clei*, *eci*, *name*, *qual*, *sln*, *ssn*.

The parameter set in the ED-EQPT (socket) Notification depends on the EEPROM format of the Optical Module which is plugged out.

Circuit pack plug out with plugged in modules

If a circuit pack will be plugged out with plugged in modules or with fixed mounted modules the system behaves like two independent actions, which will be performed one after the other: plug out of the plugged in or fixed mounted modules and plug out of the circuit pack. Regarding modules notifications will be sent as described for module plug out.

Circuit pack pre-provisioning

If a circuit pack, which supports sockets, is pre-provisioned using the ENT-EQPT command for a slot, then the creation of the socket entities shall be reported via ENT-EQPT notifications for sockets with empty spec block. Example:

```
   LT-UNITE^06-03-15 11:14:13
A  004 REPT DBCHG
   ″TIME=11-14-13,DATE=06-03-15:ENT-EQPT:1-1-#-#-39-sc1::″
;
```

Circuit pack

Overview

Purpose

This section lists the circuit pack level configuration messages.

Contents

List of commands

Autonomous notifications

System start-up with default DB

The system autonomously creates the following entities when starting up with the default database:

- backplane (backpl)
- user panel (upl)
- fan unit (fan)
- two TIs (tia and tib)
- CI-CTL (cictl)
- two PIs (pia and pib)
- CTL (ctlw-cp).

No notifications is sent.

Change of the operational state

A change of the operational state is autonomously reported by means of an ED-STATE-EQPT notification for this circuit pack.

Port

Overview

Purpose

The term “port” is defined as physical interface of the EP51, EP155 and OPn packs including the 10-Gigabit Ethernet WANPHY interface variant. The ports of the OPn packs can work in SDH or SONET mode. Therefore SDH and SONET related commands are defined for physically identical ports. The ports of the EP155 packs can only work in SDH mode. Therefore only SDH related commands are defined.

TL1 commands related to gigabit Ethernet LAN ports (GE1 packs) are described in the “Virtual switch” (p. 7-16) section.

Contents

List of commands

- "ENT-OC12 notification" (p. 3-1635)
- "DLT-OC192 notification" (p. 3-215)
- "ED-T3: Edit DS3 port" (p. 3-1202)
- "ED-T3 notification" (p. 3-1210)
- "ENT-T3 notification" (p. 3-1890)
- "DLT-T3 notification" (p. 3-258)
- "ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card" (p. 3-1306)
- "ED-TMAT3 notification" (p. 3-1310)
- "ENT-TMAT3 notification" (p. 3-1941)
- "DLT-TMAT3 notification" (p. 3-263)
- "ENT-TMST3 notification" (p. 3-1943)
- "DLT-TMST3 notification" (p. 3-264)
- "ED-EC1: Edit EC1 port" (p. 3-317)
- "ED-EC1 notification" (p. 3-325)
- "ENT-EC1 notification" (p. 3-1515)
- "DLT-EC1 notification" (p. 3-163)
- "ED-OTU2: Edit OTU2 port" (p. 3-730)
- "ED-OTU2 notification" (p. 3-736)
- "ENT-OTU2 notification" (p. 3-1688)
- "DLT-OTU2 notification" (p. 3-216)

Tributary

Overview

Purpose

The following chapter lists the commands for VC-3/STS-1 tributaries and path termination functions. The tributaries are related to the synchronous optical/electrical interfaces (e.g. on OPn or EP155). The tributaries with path termination function are related to asynchronous ports (e.g. DS3 port on EP51).

SONET and SDH tributaries are supported to keep the compatibility to current generation equipment as far as possible and to be compatible with the Telcordia manager.

Contents

List of commands

Autonomous notification

Impact of ONNS cross-connection creation

On creation of an ONNS cross-connection (*xconcls*=NN), which involves tributaries on EDGE ports (*portClass*=EDGE) which are part of a BLSR/MS-SPRING protection group the non-empty *loca*, *locz* information of the tributaries are copied into the corresponding attributes of the cross-connection and the *loca*, *locz* parameters in the tributaries are reset to its default values.

The reset of the *loca*, *locz* parameters in the tributaries are reported by means of an ED-STS (*istd* of related port SONET) or ED-VC3 (*istd* of related port SDH) notification.

6 TL1 messages - fault management

Overview

Purpose

This section lists the TL1 interface definition for the domain *fault management*.

Contents

Introduction

Overview

Purpose

This chapter presents general information necessary to understand the subsequent sections.

Contents

Alarms, events, ASAPs

Alarms/events

The system reports failures via alarms and events. An alarm/event can be persistent or transient. If it is persistent then a declaration (when the alarm will be raised) and a clear (when the alarm will be cleared) message will be sent by the NE. If it is transient only a declaration message will be sent.

Each one has a probable cause assigned. This is an identifier for the alarm/event. It will be reported via the *condtype* parameter in several commands/messages.

Alarms (conditions) with the severity Critical, Major, or Minor – will be reported via REPT ALM and can be retrieved via RTRV-ALM. Events/Alarms (conditions) with severity Warning/Not-Alarmed – will be reported via REPT EVT and can be retrieved via RTRV-COND. Alarms of environmental points will be not reported and retrieved via this command. Refer to MDI/MDO description. The different reporting messages and retrieve commands are because of TL1 requirements.

Service affecting / non service affecting

An alarm can be service affecting (sa) or non service affecting (nsa). This indicates if transmission on a path may be affected (sa) or not (nsa). It is only a well educated guess based on the limited view of a single NE.

Alarm severity

Each alarm/event will be reported with a severity. The severity specifies the urgency of the alarm/event. The severity which will be sent for a specific probable cause can be configured via ASAPs.

The NE supports the following severities:

- CR – Critical (ITU / Telcordia)
- MJ – Major (ITU / Telcordia)
- MN – Minor (ITU / Telcordia)
- NA – Warning (ITU) / Not-Alarmed (Telcordia)
- NR – Non-Alarmed (ITU) / Not-Reported (Telcordia).

ASAPs

Alarm Severity Assignment Profiles (ASAP) assign for each probable cause and if applicable service affecting and non service affecting independent a severity. Each ASAP parameter of an ASAP refers to a probable cause and if applicable for service affecting or not. There are different types of ASAPs which defines the severities for different sets of probable causes. An entity which can raise an alarm/event will be assigned an ASAP which defines for the specific alarms/events of the entity the severity. For each ASAP type a default ASAP exists in the NE, which cannot be deleted. ASAPs can be created via ENT-ASAP-PROF, edited via ED-ASAP-PROF, deleted via DLT-ASAP-PROF and retrieved via RTRV-ASAP-PROF.

SONET / SDH Mode

The NE does not distinguish between SONET and SDH. For both the same alarms will be reported and only one set of severities is supported. Alarms use probable causes where applicable from the Telcordia (″SONET world″) standard. For alarm description is system specific and not adapted to Telcordia standards.

For severities the notation is used like described in the TL1 standards of Telcordia. For explanation the meaning in the SONET and SDH world is added.

MDI / MDO

MDI

Miscellaneous Discrete Inputs (MDI / environmental point) are used to have some external input which can raise alarms like from a fire detector. The name and alarm description of an MDI can be configured. The general discussion for alarms/events is also true for MDIs. Only the alarms will be reported via REPT-ALM-ENV and can be retrieved via RTRV-ALM-ENV. The configuration of MDIs can be edited via SET-ATTR-ENV and retrieved via RTRV-ATTR-ENV.

MDO

Miscellaneous Discrete Outputs (MDO / external control) are outputs which can operate equipment outside of the NE like a lamp. Each MDO has a name which can be configured. MDOs can be configured via SET-ATTR-CONT and can be retrieved via RTRV-ATTR-CONT.

MDOs can be operated in two modes: automatically and via management commands. For the automatic mode an MDO gets assigned a probable cause. If at least one alarm of this probable cause is active then the MDO will be operated. For operation with the management commands no probable cause must be assigned to the MDO. The MDO can be operated via OPR-EXT-COND and RLS-EXT-COND. The control state of an MDO can be retrieved with the RTRV-EXT-CONT command.

Alarm assignment profiles

Overview

Purpose

This section comprises messages to handle alarm assignment profiles (ASAPs).

Contents

List of commands

Alarms

Overview

Purpose

This section lists messages to handle alarms.

Contents

List of commands

Alarm log

Overview

Purpose

This section comprises messages to handle alarm logs.

Contents

RTRV-LOG-ALM: Retrieve log alarm

For a description of this command, refer to "RTRV-LOG-ALM: Retrieve log alarm" (p. 3-2454).

MDI

Overview

Purpose

This section lists information about the TL1 commands related to MDI.

Contents

List of commands

MDO

Overview

Purpose

This section lists information about the TL1 commands related to MDO.

Contents

List of commands

Alarm, event, ASAP tables

Overview

Purpose

This section describes which alarms and events in the system can occur.

For each possible alarm following will be described:

condtype

The *condtype*, it represents the probable cause.

sa/nsa

States if it is service affecting (sa) or/and non service affecting (nsa).

conddescr

The *conddescr* related to the alarm. It is of the format:

1. \″<category>, <entity>, <description>, <additional text>\″
2. \″<category>, <entity>, <description>, <additional text>-cleared\″.

If there is no additional text, then the trailing comma may be omitted.

The *conddescr* has two forms for persistent alarms:

1. It is used when the alarm is raised.
2. The string: ″-cleared″ is appended. It is used when the alarm will be cleared. Transient alarms use only the first form.

The description in the tables below defines the text for each part of the *condescr*. The leading and ending ’ \″ ’ are omitted.

ASAP Parameter

The ASAP parameter which are used in the RTRV-ASAP-PROF, ED-ASAP-PROF, ENT-ASAP-PROF commands for the specific ASAP type.

The name of service affected related ASAP parameters starts with the prefix ″sa_″, for non service affected one with ″nsa_″. The prefix can be omitted if the one probable cause reports only for service affected or non service affected.

default severity

The default alarm severity which is assigned to the ASAP parameter.

Description

A description for explanation.

modifier/aidtype

The to the alarm related *modifier* and *aidtype.*

AID

The AIDs which can be reported related to the alarms. All wildcard aids like **`all`** must not be used.

Alarm duration

It describes if the alarm is a persistent one or a transient one.

There are separate ASAP parameter for sa and nsa alarms. There applicable each parameter will be described separately together with their default values.

The Alarms will be described per ASAP type.

Please refer also to other fragments of the TL1 documentation where REPT EVT messages related to this fragment will be defined.

Remark to TCA: The alarm level of threshold crossing alarms (TCA) shall always be “not alarmed”, and it is not user provisionable. The system shall not allow the user to change any attribute of TCA alarm through ASA profile. All the attributes, such as TCA reporting enable/disable, that will affect TCA reporting shall be handled by the TCA profile.

Contents

ASAP Type DCN

Properties of all alarms

modifier/aidtype: COM

AID: SystemAID

Alarm duration: persistent.

condtype: PARARREP

sa/nsa: nsa

conddescr: Communication-DCN, DCN, Partitioned Area Repair

ASAP parameter: *pararrep*

Default severity: MN.

condtype: PROVINVAL

Important! *Please note that this condtype is not supported by the current release.*

sa/nsa: nsa

conddescr: Communication-DCN, DCN, DCN Provisioning Not Valid

ASAP parameter: *provinval*

Default severity: MJ.

condtype: ADDROV

sa/nsa: nsa

conddescr: Communication-DCN, DCN, Address List Overflow

ASAP parameter: *addrov*

Default severity: MJ.

condtype: ASTNNBRNAV

sa/nsa: nsa

conddescr: Communication-SCN, SCN, ASTN Neighbor Not Available

ASAP parameter: *astnnbrnav*

Default severity: MJ.

modifier/aidtype: COM

AID: SystemAID

The alarm information also includes the IP address of the affected ASTN neighbor, which has to be mapped into the *conddescr*.

condtype: NSAPHIGH

sa/nsa: nsa

conddescr: Communication-DCN, DCN, NSAP counter exceeds threshold

ASAP parameter: *nsaphigh*

Default severity: MN.

condtype: BUFROVLD

sa/nsa: nsa

conddescr: Communication-DCN, DCN, NSAP counter over maximum

ASAP parameter: *bufrovld*

Default severity: MJ.

ASAP Type DS1

Properties of all alarms

modifier/aidtype: DS1

AID: DS1NumAID

Alarm duration: persistent.

condtype: DS1LOF-EGR

sa/nsa: sa

conddescr: Communication-Transport, DS1 path, DS1 LOF egress

ASAP parameter: *sa_ds1lofe* (service affecting).

default severity: NR

condtype: DS1AIS-EGR

sa/nsa: sa

conddescr: Communication-Transport, DS1 path, DS1 AIS egress

ASAP parameter: *sa_ds1aise* (service affecting).

default severity: NR

condtype: DS1RAI-EGR

sa/nsa: sa

conddescr: Communication-Transport, DS1 path, DS1 RAI egress

ASAP parameter: *sa_ds1raie* (service affecting).

default severity: NR

ASAP Type DS3

Properties of all alarms

Alarm duration: persistent

condtype: DS3LOS

sa/nsa: sa

conddescr: Communication-Transport, T3 port, DS3 Loss of Signal

ASAP parameter: *sa_ds3los* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3LineDEG

sa/nsa: sa

conddescr: Communication-Transport, T3 port, Degraded DS3 Line signal

ASAP parameter: *sa_ds3deg* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3LOF

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 Loss of Frame

ASAP parameter: *sa_ds3lof* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3AIS

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 AIS

ASAP parameter: *sa_ds3ais* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3AIC

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 Application Mismatch in

ASAP parameter: *sa_ds3aici* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3IDLE

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 IDLE in

ASAP parameter: *sa_ds3idlei* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3RAI

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 RAI in

ASAP parameter: *sa_ds3raii* (service affecting).

default severity: MJ

modifier/aidtype: T3

AID: APortAID

condtype: DS3LOF-EGR

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 LOF egress

ASAP parameter: *sa_ds3lofe* (service affecting).

default severity: NR

modifier/aidtype: T3, TMAT3

AID: APortAID, TMuxAPortAID

condtype: DS3AIS-EGR

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 AIS egress

ASAP parameter: *sa_ds3aise* (service affecting).

default severity: NR

modifier/aidtype: T3, TMAT3

AID: APortAID, TMuxAPortAID

condtype: DS3IDLE-EGR

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 IDLE egress

ASAP parameter: *sa_ds3idlee* (service affecting).

default severity: MJ

modifier/aidtype: TMAT3

AID: TMuxAPortAID

condtype: DS3RAI-EGR

sa/nsa: sa

conddescr: Communication-Transport, T3 path, DS3 RAI egress

ASAP parameter: *sa_ds3raie* (service affecting).

default severity: NR

modifier/aidtype: T3, TMAT3

AID: APortAID, TMuxAPortAID

ASAP Type ENET

Properties of all alarms

Alarm duration: persistent

condtype: VCGLOA

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Loss of Alignment

ASAP parameter: sa_vcg_loa (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_loa (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGLOPC

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Partial Transport Capacity Loss

ASAP parameter: sa_vcg_lopc (service affecting)

Default severity: MJ

ASAP parameter: nsa_vcg_lopc (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGPLCR

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Partial Loss of Capacity Rcvr

ASAP parameter: sa_vcg_plcr (service affecting)

Default severity: MJ

ASAP parameter: nsa_vcg_plcr (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGPLCT

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Partial Loss of Capacity Transm

ASAP parameter: sa_vcg_plct (service affecting)

Default severity: MJ

ASAP parameter: nsa_vcg_plct (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGLOTC

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Total Transport Capacity Loss

ASAP parameter: sa_vcg_lotc (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_lotc (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGTLCR

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Total Loss of Capacity Rcvr

ASAP parameter: sa_vcg_tlcr (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_tlcr (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGTLCT

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Total Loss of Capacity Transm

ASAP parameter: sa_vcg_tlct (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_tlct (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGFOPT

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Source End Failure of Protocol

ASAP parameter: sa_vcg_fopt (service affecting)

Default severity: MJ

ASAP parameter: nsa_vcg_fopt (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGFOPR

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Sink End Failure of Protocol

ASAP parameter: sa_vcg_fopr (service affecting)

Default severity: MJ

ASAP parameter: nsa_vcg_fopr (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: VCGSSF

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Server Signal Fail

ASAP parameter: sa_vcg_ssf

Default severity: NR

ASAP parameter: nsa_vcg_ssf

Default severity: NR

modifier/aidtype: VCG

AID: VCGAID.

condtype: GFPLOF

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, GFP Loss of Frame

ASAP parameter: sa_gfp_lof

Default severity: CR

ASAP parameter: nsa_gfp_lof

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID.

condtype: GFPLFD

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, Loss of Frame Delineation

ASAP parameter: sa_vcg_lfd (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_lfd (non service affecting)

Default severity: MN

modifier/aidtype: 1GE

AID: VCGAID.

condtype: GFPUPM

sa/nsa: sa / nsa

conddescr: Communication-Transport, VCG, GFP User Payload Mismatch

ASAP parameter: sa_vcg_upm (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_upm (non service affecting)

Default severity: MN

modifier/aidtype: VCG

AID: VCGAID

condtype: GFPEXM

sa/nsa: nsa

conddescr: Communication-Transport, VCG, GFP Extension Header Mismatch

ASAP parameter: sa_vcg_exm (service affecting)

Default severity: CR

ASAP parameter: nsa_vcg_exm (non service affecting)

Default severity: NM

modifier/aidtype: CVG

AID: VCGAID.

condtype: GFPCSF

sa/nsa: nsa

conddescr: Communication-Transport, VCG, VCG Client Signal Fail

ASAP parameter: sa_vcg_csf (service affecting)

Default severity: NR

ASAP parameter: nsa_vcg_csf (non service affecting)

Default severity: NR

modifier/aidtype: VCG

AID: VCGAID.

condtype: LANLOS

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype}, LAN Loss of Signal

ASAP parameter: sa_lan_los (service affecting)

Default severity: CR

ASAP parameter: nsa_lan_los (non service affecting)

Default severity: MN

modifier/aidtype: 1GE, 1GEPL, 10GEPL

AID: EPortAID.

condtype: LANANM

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype}, LAN Auto Negotiation Mismatch

ASAP parameter: sa_lan_anm (service affecting)

Default severity: CR

ASAP parameter: nsa_lan_anm (non service affecting)

Default severity: MN

modifier/aidtype: 1GE, 1GEPL, 10GEPL

AID: EPortAID.

condtype: MACVLANOVFW

sa/nsa: nsa

conddescr: Communication-Transport, 1GE, max no of VLAN instances reached

ASAP parameter: nsa_vlan_maxno (service affecting)

Default severity: MN

modifier/aidtype: 1GE

AID: EPortAID.

Per Gigabit Ethernet card one virtual switch is available, therefore the alarm is always raised against the first port of the card. Also the alarm configuration is related to the first port. condtype: LANPHYNOSYNC

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype}, Loss of Synchronisation Eport

ASAP parameter: sa_lan_nosync (service affecting)

Default severity: CR

ASAP parameter: nsa_lan_nosync (non service affecting)

Default severity: MN

modifier/aidtype: 1GEPL, 10GEPL

AID: EPortAID.

ASAP Type ENV

Properties of all alarms

modifier/aidtype: ENV

AID: Miscellaneous_in_DiscreteAID

Alarm duration: persistent.

Reference ASAP Type ENV

Refer to the *envmsg* parameter in the output parameters section of the "RTRV-ATTR-ENV: Retrieve attribute environment" (p. 3-2257) command for string values as part of *conddescr*.

condtype: MISC_1

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_1

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_2

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_2

Default severity: NA

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_3

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_3

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_4

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_4

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_5

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_5

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_6

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_6

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_7

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_7

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

condtype: MISC_8

sa/nsa: none

conddescr: {envmsg}

ASAP parameter: envron_8

Default severity: NR

Alarm of MDI (Miscellaneous Discrete Input / environmental point). The description can be assigned.

ASAP Type EQUIP-PROT

Properties of all alarms

modifier/aidtype: EQPT

Alarm duration: persistent

condtype: EQPT

sa/nsa: sa / nsa

conddescr: Equipment, {cpname}/{cpqual}, Circuit Pack Clock Failure

ASAP parameter: sa_eqpt;

Default severity: CR

ASAP parameter: nsa_eqpt;

Default severity: MN

AID: CpAID.

condtype: CPTEMP

sa/nsa: sa / nsa

conddescr: Equipment, {cpname}/{cpqual}, Unit Temperature too High

ASAP parameter: sa_cptemp;

Default severity: MJ

ASAP parameter: nsa_cptemp;

Default severity: MN

AID: CpAID.

condtype: PRCDRERR

sa/nsa: sa / nsa

conddescr: Equipment, slot, Circuit Pack Type Mismatch

ASAP parameter: sa_prcdrerr;

Default severity: MJ

ASAP parameter: nsa_prcdrerr;

Default severity: MN

AID: SlotAID.

condtype: REPLUNITMISS

sa/nsa: sa / nsa

conddescr: Equipment, slot, Circuit Pack not Present

ASAP parameter: sa_replunitmiss;

Default severity: CR

ASAP parameter: nsa_replunitmiss;

Default severity: MN

AID: SlotAID.

condtype: TXInF

sa/nsa: sa / nsa

conddescr: Equipment, {cpname}/{cpqual}, TXI Failure

ASAP parameter: sa_txif;

Default severity: CR

ASAP parameter: nsa_txif;

Default severity: MN

AID: CpAID.

Description: The n in the condtype expresses the individual TXI line for which the alarm is reported. The maximum of n depends on the card type. The condtype will be reported as: TXI1F, TXI2F.

condtype: CPFAIL

sa/nsa: sa / nsa

conddescr: Equipment, {cpname}/{cpqual}, Circuit Pack Failure

ASAP parameter: sa_cpfail;

Default severity: CR

ASAP parameter: nsa_cpfail;

Default severity: MN

AID: CpAID.

condtype: PIMISS

sa/nsa: sa / nsa

conddescr: Equipment, shelf, Power Interface not Present

ASAP parameter: sa_pimiss;

Default severity: MJ

ASAP parameter: nsa_pimiss;

Default severity: MN

AID: PowerAID.

condtype: PIEF

sa/nsa: nsa

conddescr: Equipment, shelf, Power Interface Read Failure

ASAP parameter: nsa_pief;

Default severity: MN

AID: PowerAID.

condtype: POWF

sa/nsa: sa / nsa

conddescr: Equipment, shelf, System Power Failure

ASAP parameter: sa_powf;

Default severity: MJ

ASAP parameter: nsa_powf;

Default severity: MN

AID: PowerAID.

condtype: ONIFW

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, ONI Failure on working CTL slot

ASAP parameter: sa_onifw;

Default severity: MN

AID: CpAID.

condtype: ONIFP

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual},ONI Failure on protecting CTL

ASAP parameter: nsa_onifp;

Default severity: MN

AID: CpAID.

condtype: DCTLUNAVAIL

sa/nsa: nsa

conddescr: Equipment, SYSTEM, Duplex Control not Present

ASAP parameter: nsa_dctlunavail;

Default severity: NR

AID: SystemAID.

condtype: BKUPMEMP

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, IDE Flash Card Access Fail

ASAP parameter: nsa_bkupmemp

Default severity: MJ

AID: CpAID

condtype: OMMISMA

sa/nsa: sa/nsa

conddescr: Equipment, Socket, Optical Module Type Mismatch

ASAP parameter: sa_ommisma

Default severity: CR

ASAP parameter: nsa_ommisma

Default severity: MN

AID: SocketAID.

condtype: OMMISS

sa/nsa: sa/nsa

conddescr: Equipment, Socket, Optical Module not Present

ASAP parameter: sa_ommiss

Default severity: CR

ASAP parameter: nsa_ommiss

Default severity: MN

AID: SocketAID.

condtype: OMFAIL

sa/nsa: sa/nsa

conddescr: Equipment, Socket, Optical Module Fail

ASAP parameter: sa_omfail

Default severity: CR

ASAP parameter: nsa_omfail

Default severity: MN

AID: SocketAID.

condtype: OMUNSUP

sa/nsa: sa/nsa

conddescr: Equipment, Socket, Optical Module not supported

ASAP parameter: sa_omunsup

Default severity: CR

ASAP parameter: nsa_omunsup

Default severity: MN

AID: SocketAID.

condtype: OPTFREQnMIS

sa/nsa: sa/nsa

conddescr: Equipment, {cpname}/{cpqual}, Optical WDM Frequency Mismatch

ASAP parameter: sa_optfreqmis

Default severity: CR

ASAP parameter: nsa_optfreqmis

Default severity: MN

AID: CpAID.

The n in the condtype expresses the number of the port, for which the alarm is reported.

ASAP Type EQUIP-UNPROT

Properties of all alarms

modifier/aidtype: EQPT

Alarm duration: persistent

condtype: CSDWF

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, Worker Clock Input Fail

ASAP parameter: nsa_csdwf;

Default severity: MN

AID: CpAID.

condtype: CSDPF

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, Protection Clock Input Fail

ASAP parameter: nsa_csdpf;

Default severity: MN

AID: CpAID.

condtype: SYNCCLK

sa/nsa: sa

conddescr: Equipment, system, NE Clock Failure

ASAP parameter: sa_syncclk;

Default severity: CR

AID: SystemAID.

condtype: CPDEGR

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, Unit Cooling Degraded

ASAP parameter: nsa_cpdegr;

Default severity: MN

AID: CpAID.

condtype: CPCOMF

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, Circuit Pack Comm Failure

ASAP parameter: nsa_cpcomf;

Default severity: MN

AID: CpAID.

condtype: BPEF

sa/nsa: sa

conddescr: Equipment, shelf, Backplane Read Failure

ASAP parameter: sa_bpef;

Default severity: MJ

AID: ShelfAID.

condtype: SYSCOMF

sa/nsa: sa

conddescr: Equipment, {cpname}/{cpqual}, Comm Channel Failure

ASAP parameter: sa_syscomf default; severity: MJ

AID: CpAID.

condtype: CTLCOMF

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, CTL Comm Failure

ASAP parameter: nsa_ctlcomf;

Default severity: MN

AID: CpAID.

condtype: BKUPMEMO

sa/nsa: nsa

conddescr: Processing Error, {cpname}/{cpqual}, NVM Shortage

ASAP parameter: nsa_bkupmemo;

Default severity: MN

AID: CpAID.

condtype: CIPWRDWN

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, CICTL Powered Down

ASAP parameter: nsa_cipwrdwn;

Default severity: MN

AID: CpAID.

condtype: TIMM

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, TI Mismatch

ASAP parameter: nsa_timm;

Default severity: MN

AID: CpAID.

condtype: TIMISS

sa/nsa: nsa

conddescr: Equipment, {cpname}/{cpqual}, Shelf, TI not Present

ASAP parameter: nsa_timiss

Default severity: MN

AID: TimingInterfaceAID.

condtype: TIMEF

sa/nsa: nsa

conddescr: Equipment, TI, TI EEPROM Access Failure

ASAP parameter: nsa_timef;

Default severity: MN

AID: TimingInterfaceAID.

condtype: CIMISS

sa/nsa: nsa

conddescr: Equipment, shelf, CICTL not Present

ASAP parameter: nsa_cimiss;

Default severity: MN

AID: ShelfAID.

condtype: CIF

sa/nsa: nsa

conddescr: Equipment, ci-ctl, CICTL Failure

ASAP parameter: nsa_cif;

Default severity: MN

AID: CiCtlAID.

condtype: CICOMF

sa/nsa: nsa

conddescr: Equipment, ci-ctl, CICTL Comm Failure

ASAP parameter: nsa_cicomf;

Default severity: MN

AID: CiCtlAID.

condtype: FANUNITMISS

sa/nsa: sa

conddescr: Equipment, shelf, Fan Unit not Present

ASAP parameter: sa_fanunitmiss;

Default severity: CR

AID: ShelfAID.

condtype: FANUNITCOMF

sa/nsa: nsa

conddescr: Equipment, fan, Fan Unit Comm Failure

ASAP parameter: nsa_fanunitcomf;

Default severity: MN

AID: FanAID.

condtype: FANI2CALM

sa/nsa: nsa

conddescr: Equipment, fan, FAN 2 I2C Failure

ASAP parameter: nsa_fani2calm;

Default severity: MN

AID: FanAID.

condtype: FANUNITF

sa/nsa: sa

conddescr: Equipment, fan, Fan Unit Failure

ASAP parameter: sa_fanunitf

Default severity: CR

AID: FanAID.

condtype: FANF

sa/nsa: sa

conddescr: Equipment, fan, Fan Failure

ASAP parameter: sa_fanf;

Default severity: MJ

AID: FanAID.

condtype: FEEDAF

sa/nsa: nsa

conddescr: Equipment, fan, Fan Voltage Feed A Failure

ASAP parameter: nsa_feedaf;

Default severity: MN

AID: FanAID.

condtype: FEEDBF

sa/nsa: nsa

conddescr: Equipment, fan, Fan Voltage Feed B Failure

ASAP parameter: nsa_feedbf;

Default severity: MN

AID: FanAID.

condtype: UPMISS

sa/nsa: nsa

conddescr: Equipment, shelf, User Panel not Present

ASAP parameter: nsa_upmiss;

Default severity: MN

AID: ShelfAID.

condtype: UPCOMF

sa/nsa: nsa

conddescr: Equipment, user panel, User Panel Comm Failure

ASAP parameter: nsa_upcomf;

Default severity: MN

AID: UserPanelAID.

condtype: ECIMISMA

sa/nsa: nsa

conddescr: Equipment, ECI, ECI Mismatch Failure

ASAP parameter: sa_ecimisma;

Default severity: CR

AID: EciAID.

condtype: ECICOMF

sa/nsa: nsa

conddescr: Equipment, ECI, ECI Comm Failure

ASAP parameter: nsa_ecicomf;

Default severity: MN

AID: EciAID.

condtype: ECIMISS

sa/nsa: sa

conddescr: Equipment, ECI, ECI not Present

ASAP parameter: sa_ecimiss;

Default severity: CR

AID: EciAID.

ASAP Type MSSPRING-BLSR

Properties of all alarms

modifier/aidtype: OC48, OC192, OC768, STM16, STM64, STM256.

Alarm duration: persistent.

condtype: APSC

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Inconsistent APS Codes

ASAP parameter: inaps;

Default severity: MJ

AID: SPortAID (2F: west/east port, 4F: west/east protection port).

condtype: IMAPS

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Improper APS Codes

ASAP parameter: imaps;

Default severity: MJ

AID: SPortAID (2F: west/east port, 4F: west/east protection port).

condtype: NID-CONFL

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Node ID Mismatch

ASAP parameter: nid_confl;

Default severity: MJ

AID: SPortAID (2F: west/east port, 4F: west/east protection port).

condtype: DKB

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Default K-bytes

ASAP parameter: dkb;

Default severity: MJ

AID: SPortAID (2F: west/east port, 4F: west/east protection port).

condtype:RNG-PSS

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Ring protection switch Suspended

ASAP parameter: rng_pss;

Default severity: MJ

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: TS

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Traffic Squelched

ASAP parameter: ts;

Default severity: MJ

AID: SPortAID (2F: west/east port, 4F: west/east worker port).

condtype: DUPL-RNG

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Duplicate Ring Node

ASAP parameter: dupl_rng;

Default severity: MJ

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: RNG-OPEN

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Open Ring

ASAP parameter: ropn;

Default severity: MJ

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: SQMAP-CONFL

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Local Squelch Map Conflict

ASAP parameter: sqmap_confl;

Default severity: MN

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: RNG-INC

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Ring Incomplete

ASAP parameter: rng_inc;

Default severity: MJ

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: RNG-URT

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Unknown Ring Type

ASAP parameter: rng_urt;

Default severity: NA

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: RNG-IRPM

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Inconsistent Ring Protection Mode

ASAP parameter: rng_irpm;

Default severity: MJ

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: RNG-DSCVY

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Ring Discovery in Progress

ASAP parameter: rng_dscvy;

Default severity: NA

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID.

condtype: ET-PREEMPT

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Extra Traffic Preempted

ASAP parameter: et_preempt;

Default severity: NA

AID: SPortAID (2F: west/east port, 4F: west/east protection port).

condtype: RINGCKTVALSUSPEND

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Ring Ckt Validation Suspended

ASAP parameter: ringcktvalsuspend;

Default severity: MN

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID

condtype: PIF

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Path Integrity Failure

ASAP parameter: pif;

Default severity: MJ

AID: STS1NumAID

condtype: RINGCKTALARMSUPPR

sa/nsa: nsa

conddescr: Communication-Transport, MSSPRING-BLSR prot group, Ring Circuit Alarms Suppressed

ASAP parameter: ringcktalarmsuppr;

Default severity: NA

AID: TwoFibre_ProtGrpAID, FourFibre_ProtGrpAID

ASAP Type MSP-APS

Properties of all alarms

modifier/aidtype: OC3, OC12, OC48, OC192, OC768, STM1, STM4, STM16, STM64, STM256

AID: OnePlusOne_PrtGrpAID

Alarm duration: persistent.

condtype: PAMARCH

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Prot. Arch. Mismatch

ASAP parameter: pamarch;

Default severity: MJ.

condtype: PAMOPER

sa/nsa: nsa

conddescr: Communication-Transport, 1+1 MSP-APS prot group, Prot. Arch. Operation Mismatch

ASAP parameter: pamoper;

Default severity: MJ.

condtype: PAMMODE

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Prot. Arch. Mode Mismatch

ASAP parameter: pammode;

Default severity: MJ.

condtype: PSBUNACC

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Prot. Switch Byte Unacceptable

ASAP parameter: psbunacc;

Default severity: MJ.

condtype: PSBINAPPR

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Prot. Switch Byte Inappropriate

ASAP parameter: psbinappr;

Default severity: MJ.

condtype: SCM

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Switch Channel Mismatch

ASAP parameter: scm;

Default severity: MJ.

condtype: PRIM

sa/nsa: nsa

conddescr: Communication-Transport, 1+1 MSP-APS prot group, primary section Mismatch

ASAP parameter: prim;

Default severity: MJ.

condtype: FESF

sa/nsa: nsa

conddescr: Communication-Transport, MSP-APS prot group, Far End Signal Fail

ASAP parameter: fesf;

Default severity: MJ.

ASAP Type OTN

Properties of all alarms

Alarm duration: persistent

Modifier/aidtype: OTU2.

AID: OTNPortAID

If properties are explicitely specified for an alarm, then the properties specified at the alarm are valid.

condtype: OCHLOS

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Loss of Payload of OTN Channel

ASAP parameter: sa_ochlos;

Default severity: CR

ASAP parameter: nsa_ochlos

Default severity: MN.

condtype: OTULOF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, OTUk Loss of Frame

ASAP parameter: sa_otulof;

Default severity: CR

ASAP parameter: nsa_otulof;

Default severity: MN.

condtype: OTULOM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, OTUk Loss of Multiframe

ASAP parameter: sa_otulom;

Default severity: CR

ASAP parameter: nsa_otulom;

Default severity: MN.

condtype: OTUTIM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, OTU2 Trace Identifier Mismatch

ASAP parameter: sa_ototim;

Default severity: CR

ASAP parameter: nsa_ototim;

Default severity: MN.

condtype: OTUBDI

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, OTUk Backward Defect Indication

ASAP parameter: sa_otubdi;

Default severity: NR

ASAP parameter: nsa_otubdi;

Default severity: NR.

ASAP Type PATH

Properties of all alarms

modifier/aidtype: VTL, STS3C, STS12C, STS48C, STS192C, VC12, C4, VC44C, VC416C, VC464C.

AID: LONumAID, TS1NumAID

Alarm duration: persistent.

condtype: AIS-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Alarm Indication Signal

ASAP parameter: sa_ais;

Default severity: NR

ASAP parameter: nsa_ais;

Default severity: NR.

condtype: LOP-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Loss of Pointer

ASAP parameter: sa_lop;

Default severity: CR

ASAP parameter: nsa_lop;

Default severity: MN.

condtype: SRM-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Signal Rate Mismatch

ASAP parameter: sa_srm;

Default severity: MJ

ASAP parameter: nsa_srm;

Default severity: MN.

condtype: UNEQ-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Unequipped

ASAP parameter: sa_uneq;

Default severity: CR

ASAP parameter: nsa_uneq;

Default severity: MN.

condtype: TIM-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Trace Identifier Mismatch

ASAP parameter: sa_tim;

Default severity: CR

ASAP parameter: nsa_tim;

Default severity: MN.

condtype: EXC-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Excessive Bit Error Ratio

ASAP parameter: sa_exc;

Default severity: CR

ASAP parameter: nsa_exc;

Default severity: MN.

condtype: DEG-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Degraded Signal

ASAP parameter: sa_deg;

Default severity: MJ

ASAP parameter: nsa_deg;

Default severity: MN.

condtype: SSF-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Server Signal Fail

ASAP parameter: sa_ssf;

Default severity: NR

ASAP parameter: nsa_ssf;

Default severity: NR.

condtype: RFI-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Remote Defect Indication

ASAP parameter: sa_rfi;

Default severity: NR

ASAP parameter: nsa_rfi;

Default severity: NR.

condtype: PDI-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Payload Defect Indication

ASAP parameter: sa_pdi;

Default severity: MJ

ASAP parameter: nsa_pdi;

Default severity: MN.

ASAP Type PATHTERM

Properties of all alarms

Alarm duration: persistent.

condtype: THPSSF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Server Signal Fail

ASAP parameter: sa_thpssf;

Default severity: NR

ASAP parameter: nsa_thpssf;

Default severity: NR.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPLOP

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Loss of Pointer

ASAP parameter: sa_thplop;

Default severity: CR

ASAP parameter: nsa_thplop;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPUNEQ

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Unequipped

ASAP parameter: sa_thpuneq;

Default severity: CR

ASAP parameter: nsa_thpuneq;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPTIM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Trace Identifier Mismatch

ASAP parameter: sa_thptim;

Default severity: CR

ASAP parameter: nsa_thptim;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPEXC

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Excessive Bit Error Ratio

ASAP parameter: sa_thpexc;

Default severity: CR

ASAP parameter: nsa_thpexc;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPDEG

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Degraded Signal

ASAP parameter: sa_thpdeg;

Default severity: MJ

ASAP parameter: nsa_thpdeg;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPRDI

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Remote Defect Indication

ASAP parameter: sa_thprdi;

Default severity: NR

ASAP parameter: nsa_thprdi;

Default severity: NR.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: THPPLM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Payload Mismatch

ASAP parameter: sa_thpplm;

Default severity: CR

ASAP parameter: nsa_thpplm;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4, TMASTS1

AID: VCGNumAID, PTNumAID, TMuxAPTNumAID

condtype: VCLOM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Loss of Multiframe

ASAP parameter: sa_vclom;

Default severity: CR

ASAP parameter: nsa_vclom;

Default severity: MN.

modifier/aidtype: STS1, STS3C, STS192C, VC4, VC464C, HOSTS1, HOVC4

AID: VCGNumAID, PTNumAID

condtype: VCSQM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Sequence Number Mismatch

ASAP parameter: sa_vcsqm;

Default severity: CR

ASAP parameter: nsa_vcsqm;

Default severity: MN.

modifier/aidtype: STS1, STS3C, VC4

AID: VCGNumAID

condtype: PDI-P

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} path, Payload Defect Indication

ASAP parameter: sa_pdi;

Default severity: MJ

ASAP parameter: nsa_pdi;

Default severity: MN.

modifier/aidtype: STS1,TMASTS1

AID: PTNumAID, TMuxAPTNumAID

condtype: LCASVCGMND

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Member not deskewable

ASAP parameter: sa_vcmnd (service affecting);

Default severity: MN

ASAP parameter: nsa_vcmnd (non service affecting);

Default severity: NR.

modifier/aidtype: STS1, STS3C, VC4

AID: VCGNumAID

condtype: TLPAIS

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Alarm Indication Signal

ASAP parameter: sa_tlpais

Default severity: NR

ASAP parameter: nsa_tlpais

Default severity: NR

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPLOP

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Loss of Pointer

ASAP parameter: sa_tlplop

Default severity: CR

ASAP parameter: nsa_tlplop

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPSSF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Server Signal Fail

ASAP parameter: sa_tlpssf

Default severity: NR

ASAP parameter: nsa_tlpssf

Default severity: NR

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPRDI

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Remote Defect Indication

ASAP parameter: sa_tlprdi

Default severity: NR

ASAP parameter: nsa_tlprdi

Default severity: NR

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPPLM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Payload Mismatch

ASAP parameter: sa_tlpplm

Default severity: CR

ASAP parameter: nsa_tlpplm

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPUNEQ

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Unequipped

ASAP parameter: sa_tlpuneq

Default severity: CR

ASAP parameter: nsa_tlpuneq

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPTIM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Trace Identifier Mismatch

ASAP parameter: sa_tlptim

Default severity: CR

ASAP parameter: nsa_tlptim

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPEXC

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Excessive Bit Error Ratio

ASAP parameter: sa_tlpexc

Default severity: CR

ASAP parameter: nsa_tlpexc

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

condtype: TLPDEG

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} CS, Degraded Signal

ASAP parameter: sa_tlpdeg

Default severity: MJ

ASAP parameter: nsa_tlpdeg

Default severity: MN

modifier/aidtype: TMSVT

AID: TMuxSLONumAID

ASAP Type PORT

Properties of all alarms

modifier/aidtype: EC1, OC3, OC12, OC48, OC192, OC768, STM1E, STM1, STM4, STM16, STM64, STM256, OC48T, STM16T (for fully transparent ports).

AID: SPortAID

Alarm duration: persistent.

condtype: OCHLOF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, OCh Loss of Frame

ASAP parameter: sa_ochlof;

Default severity: CR

ASAP parameter: nsa_ochlof;

Default severity: MN.

condtype: LOS

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Loss of Signal

ASAP parameter: sa_los;

Default severity: CR

ASAP parameter: nsa_los;

Default severity: MN.

condtype: LOF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Loss of Frame

ASAP parameter: sa_lof;

Default severity: CR

ASAP parameter: nsa_lof;

Default severity: MN.

condtype: LOF-T-EGR

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} port, Loss of Frame transp ch egress

ASAP parameter: sa_loftegr;

Default severity: CR

condtype: RSTIM

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Trace Identifier Mismatch

ASAP parameter: sa_rstim;

Default severity: CR

ASAP parameter: nsa_rstim;

Default severity: MN.

condtype: TIM-T-EGR

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} port, Trace Identifier Mismatch egress

ASAP parameter: sa__timtegr;

Default severity: CR

condtype: MSEXC

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Excessive Bit Error Ratio

ASAP parameter: sa_msexc;

Default severity: CR

ASAP parameter: nsa_msexc;

Default severity: MN.

condtype: MSDEG

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Degraded Signal

ASAP parameter: sa_msdeg;

Default severity: MJ

ASAP parameter: nsa_msdeg;

Default severity: MN.

condtype: AIS-L

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Alarm Indication Signal

ASAP parameter: sa_aisl;

Default severity: CR

ASAP parameter: nsa_aisl;

Default severity: MN.

condtype: MSSSF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Server Signal Fail

ASAP parameter: sa_msssf;

Default severity: NR

ASAP parameter: nsa_msssf;

Default severity: NR.

condtype: MSPSSF

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Prot. Server Signal Fail

ASAP parameter: sa_mspssf;

Default severity: NR

ASAP parameter: nsa_mspssf;

Default severity: NR.

condtype: RFI-L

sa/nsa: sa / nsa

conddescr: Communication-Transport, {aidtype} port, Remote Defect Indication

ASAP parameter: sa_rfil;

Default severity: NR

ASAP parameter: nsa_rfil;

Default severity: NR.

condtype: DCCRSF

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype} port, DCC failure

ASAP parameter: sa_dccrsf;

Default severity: MJ.

condtype: LIDRSM

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype} port, Protocol Version Mismatch

ASAP parameter: nsa_lidrsm

Default severity: MJ.

condtype: DCCMSF

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype} port, DCC failure

ASAP parameter: nsa_dccmsf;

Default severity: MJ.

condtype: LIDMSM

sa/nsa: nsa

conddescr: Communication-Transport, {aidtype} port, Protocol Version Mismatch

ASAP parameter: nsa_lidmsm;

Default severity: MJ.

ASAP Type SCN

Properties of all alarms

Alarm duration: persistent.

condtype: TCP1+1FAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, TCP1+1 Failure

ASAP parameter: tcp1plus1fail

Default severity: CR.

modifier/aidtype: LINK

AID: LinkAID

condtype: TCP1+1DEG

sa/nsa: nsa

conddescr: Communication-SCN, LINK, TCP1+1 Degraded

ASAP parameter: tcp1plus1deg

Default severity: MJ.

modifier/aidtype: LINK

AID: LinkAID

condtype: UNICCLANFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, UNI LAN Control Channel Fail

ASAP parameter:unicclanfail

Default severity: CR.

modifier/aidtype: LINK

AID: UNILANCCLinkAID

condtype: UNICCRSFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, UNI RS Control Channel Fail

ASAP parameter: uniccrsfail

Default severity: CR.

modifier/aidtype: LINK

AID: SPortAID

condtype: UNICCMSFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, UNI MS Control Channel Fail

ASAP parameter: uniccmsfail

Default severity: CR.

modifier/aidtype: LINK

AID: SPortAID

condtype: ENNICCLANFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, ENNI LAN Control Channel Fail

ASAP parameter: enniccla nfail

Default severity: CR.

modifier/aidtype: LINK

AID: ENNILANCCLinkAID

condtype: ENNICCRSFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, ENNI RS Control Channel Fail

ASAP parameter: enniccrsfail

Default severity: CR.

modifier/aidtype: LINK

AID: SPortAID

condtype: ENNICCMSFAIL

sa/nsa: nsa

conddescr: Communication-SCN, LINK, ENNI MS Control Channel Fail

ASAP parameter: enniccmsfail

Default severity: CR.

modifier/aidtype: LINK

AID: SPortAID

condtype: DUPROUTCONT

sa/nsa: nsa

conddescr: Communication-SCN, SCN, Duplicate Level 1 Routing Contr

ASAP parameter: duproutcont

Default severity: CR.

modifier/aidtype: COM

AID: SystemAID

condtype: RA_IDUNKNOWN

sa/nsa: nsa

conddescr: Communication-SCN, LINK, Remote Area ID Unknown

ASAP parameter: raidunknown

Default severity: MJ.

modifier/aidtype: LINK

AID: RC_Neighbor_AID

condtype: RCDEAD

sa/nsa: nsa

conddescr: Communication-SCN, LINK, Neighbor RC Dead

ASAP parameter: rcdead

Default severity: CR.

modifier/aidtype: LINK

AID: RC_Neighbor_AID

ASAP Type SNCP-UPSR

Properties of all alarms

modifier/aidtype: VT1, STS1, STS3C, STS12C, STS48C, STS192C, VC12, VC3, VC4, VC44C, VC416C, VC464C, TMASTS1

AID: LONumAID, STS1NumAID, TMuxAPTNumAID

Alarm duration: persistent.

condtype: PSI

sa/nsa: nsa

conddescr: Communication-Transport, Path prot group, Path Switch Inhibit

ASAP parameter: psinh;

Default severity: NA.

condtype: FAILTOSW

sa/nsa: nsa

conddescr: Communication-Transport, Path prot group, Path Switch Denial

ASAP parameter: psfail;

Default severity: NA.

ASAP Type TIMING

Properties of all alarms

Alarm duration: persistent

condtype: SYNCRF

sa/nsa: none

conddescr: Communication-Synchronisation, timing reference, Timing Reference Failure

ASAP parameter: syncrf;

Default severity: MJ

modifier/aidtype: SYST

AID: External_timing_ref_AID, Line_timing_ref_AID.

condtype: SYNCLOSS

sa/nsa: nsa

conddescr: Communication-Synchronisation, system timing, Loss of Synchronisation

ASAP parameter: syncloss;

Default severity: MJ

modifier/aidtype: SYST

AID: SystemAID.

condtype: SYNCT4OUT

sa/nsa: nsa

conddescr: Communication-Synchronisation, timing out, T4 quality unsufficient

ASAP parameter: synct4out;

Default severity: MJ

modifier/aidtype: OUTT

AID: External_Timing_output_port_AID.

condtype: NEFASTLOCK

sa/nsa: nsa

conddescr: Communication-Synchronisation, system timing, Fast Lock mode entered

ASAP parameter: nefastlock;

Default severity: NR

modifier/aidtype: SXST

AID: SystemAID.

ASAP Type Transparency

Properties of all alarms

modifier/aidtype: ODU1

AID: VCGAID

Alarm duration: persistent.

condtype: ODULOFT

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} port, LOF Transparent Ch

ASAP parameter: sa_oduloft.

Default severity: CR.

Description: Besides {aidtype} the entity port is used to distinguish from the ODUkP functionality.

condtype: ODULOMT

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} path, Loss of Multiframe Transparent

ASAP parameter: sa_odulomt.

Default severity: CR.

condtype: ODUBDIT

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} path, Backw Defect Transparent Ch

ASAP parameter: sa_odubdit.

Default severity: MJ.

condtype: ODUSSFT

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} path, Server Signal Fail Transparent

ASAP parameter: sa_odussft.

Default severity: NR.

condtype: ODUPLMT

sa/nsa: sa

conddescr: Communication-Transport, {aidtype} path, Payload Mismatch Transparent Ch

ASAP parameter: sa_oduplmt.

Default severity: CR.

Entity – ASAP relation

Introduction

This section describes which entity is related to which ASAP.

Entities

For description of the entities see in the fragments of the TL1 message description which defines and describes the entities. (Example: The slot entity is described in the configuration fragment.)

The entity of an alarm can be retrieved from the entity in the *condescr* in the ASAP descriptions above so it is not explicitly listed here.

If an entity has a relation to at least one ASAP does it mean that this entity will raise alarms for this entity.

Entity – ASAP relation

For each ASAP type must exist one ASAP pointer for the entity. Each ASAP pointer is related to only one ASAP type. That means that for some entities more than one ASAP pointer exist.

Fragment	Entity	ASAP Type	Entity containing ASAP pointer
Administration & Security	Session	none	none
Administration & Security	NE Security Administration	none	none
Administration & Security	User Administration	none	none
Administration & Security	Security Activity Log	none	none
Configuration Management	System	EQUIP-UNPROT	System
Configuration Management	Bay	none	none
Configuration Management	Shelf	EQUIP-PROT	Shelf
		EQUIP-UNPROT	
Configuration Management	Slot	EQUIP-PROT	Slot
Configuration Management	Circuit Pack	EQUIP-PROT	CP
		EQUIP-UNPROT	
Configuration Management	Socket	EQUIP-PROT	Socket
Configuration Management	Port	PORT	Port
	(OC3, OC12, OC48, OC192, OC768, STM1, STM4, STM16, STM64, STM256)		

Fragment	Entity	ASAP Type	Entity containing ASAP pointer
Configuration Management	Port (OC48T, STM16T)	PORT	Port
Configuration Management	Port (T3)	DS3	Port
Configuration Management	TransMUX Asynchronous Port (TMAT3)	DS3	Port
Configuration Management	Port (EC1)	PORT	Port
Configuration Management	Port (OTU2)	OTN	Port
Configuration Management	Tributary (STS1, STS3C, STS12C, STS48C, STS192C, VC3, VC4, VC44C, VC416C, VC464C)	PATH	Tributary (Configuration Management)
Configuration Management	HO Path Termination (STS1, HOSTS1, HOVC4, TMASTS1)	PATHTERM	Path Termination
Fault Management	Alarm Assignment Profiles	none	none
Fault Management	Alarms	none	none
Fault Management	Alarm Log	none	none
Fault Management	MDI	ENV	MDI
Fault Management	MDO	none	none
Transmission Cross-Connect Handling	Crossconnection	none	none
Transmission Cross-Connect Handling	Tributary	none	none
Transmission Cross-Connect Handling	Lower-order Tributary (VT1.5, VC3, VC12)	PATH	Lower-order Tributary
Transmission Cross-Connect Handling	Port	none	none
Transmission Cross-Connect Handling	User Byte - Port	none	none
Transmission Cross-Connect Handling	Ethernet Port	ENET	Ethernet Port
Transmission Cross-Connect Handling	Virtual Concatenation Group	ENET	Virtual Concatenation Group
Transmission Cross-Connect Handling	Virtual Concatenation Group	TRANSPARENCY	Virtual Concatenation Group / ODU Layer

Fragment	Entity	ASAP Type	Entity containing ASAP pointer
Transmission Cross-Connect Handling	Virtual Concatenation Group Tributary	PATHTERM	Virtual Concatenation Group Tributary
	(STS1, VC4)		
Transmission Cross-Connect Handling	LO Path Termination	PATHTERM	LO Path Termination
	(TMSVT)		
Transmission Cross-Connect Handling	DS1 NIM	DS1	DS1 NIM
	(DS1)		
Protection	Protection Group	none	none
	1+1 Equipment Protection		
Protection	Protection Group	MSP-APS	Protection Group
	1+1 Multiplex Section / Line Protection		1+1 Multiplex Section / Line Protection
Protection	Protection Group	MSSPRING-BLSR	Protection Group
	2-fibre MS-SPRing / BLSR		2-fibre MS-SPRing / BLSR
Protection	Protection Group	MSSPRING-BLSR	Protection Group
	4-fibre MS-SPRing / BLSR		4-fibre MS-SPRing / BLSR
Protection	Protection Group	SNCP-UPSR	Protection Group
	Path Protection		Path Protection
Protection	Protection Log	none	none
Protection	Ring Map	none	none
Synchronization	System Timing	TIMING	System Timing
Synchronization	Output Timing	none	none
Synchronization	Assigned Timing Reference	TIMING	Assigned Timing Reference
Synchronization	External Timing Input	none	none
Synchronization	External Timing Output	TIMING	External Timing Output
Software & Database Management	n.a	none	none
Data Communications Network	General DCN	DCN	General DCN
Data Communications Network	DCC Termination	none	none
Data Communications Network	LAN	none	none
Data Communications Network	OSI Network Layer	none	none
Data Communications Network	OSI Transport Layer and TARP	none	none
Data Communications Network	TCP / IP	none	none

Fragment	Entity	ASAP Type	Entity containing ASAP pointer
Data Communications Network	IP Map	none	none
Data Communications Network	IP Tunnel	none	none
Data Communications Network	MAP	none	none
Performance Monitoring	none	none	none
Configuration Management	ci-ctl	EQUIP-UNPROT	System
Configuration Management	fan	EQUIP-UNPROT	System
Configuration Management	user panel	EQUIP-UNPROT	System
Configuration Management	power interface	EQUIP-PROT	CP
Configuration Management	ECI	EQUIP-PROT	CP
Configuration Management	Timing Interface	EQUIP-UNPROT	CP
Signaling Communications Network	General SCN	SCN	General SCN

7 TL1 messages - transmission and cross-connect handling

Overview

Purpose

This section lists the TL1 interface definition for the

- high-order and low-order transmission domain
- user byte access functionality
- Gigabit Ethernet virtual switch configuration
- Gigabit Ethernet port (GEn) configuration
- virtual concatenation (group and tributary) configuration for GEn applications
- virtual concatenation (group and tributary) configuration for ES64U applications
- high-order and low-order remote test access function.

Contents

Cross-connection

Overview

Purpose

The cross-connection (CRS) commands cover all type of

- creating (ENT-CRS),
- removing (DLT-CRS),
- modifying (ED-CRS) and
- retrieving (RTRV-CRS) cross-connections.

Furthermore the commands ENT-ROLL directly affect existing cross-connections by rolling the input of an existing cross-connection towards another input while keeping the output unchanged.

Cross-connections and tributaries

Physically cross-connections are transport channels between tributaries within an NE. These cross-connections are logically modeled by the entity cross-connection and include provisionable parameters which describe the behavior of these channels.

Tributaries are path-level units of bandwidth within ports. With the introduction of the LOXC feature, there has been a second level of tributary added: Low-order tributaries are contained in high-order tributaries within ports.

These tributaries may belong

- either as high-order tributaries to SDH/SONET ports (for example, located on OP***n***), which are to be addressed via the STS1NumAIDs,
- or as high-order tributaries to Virtual Concatenation Group ports (for example, located on GE***n***) which are to be addressed via the VCGNumAIDs,
- or as high-order path terminations to asynchronous ports (for example, located on EP51) which are to be addressed via the STS1NumAIDs,
- or as high-order path terminations to adjunct ports (for example, located on TransMUX pack) which are to be addressed via TMuxAPTNumAIDs
- or as low-order tributaries to high-order tributaries of SDH/SONET ports (for example, located on OP***n***) which are to be addressed via the LONumAIDs
- or as low-order tributaries to high-order path terminations of adjunct ports (for example, located on TransMUX pack) which are to be addressed via theTMuxSLONumAIDs.

In the following, the general AID GenTribAID is used to reference all connectable high-order tributaries and path terminations.

In the following, the general AID GenLOTribAID is used to reference all connectable low-order tributaries.

High-order tributary-rate provisioning versus cross-connection-rate

For fixed-rate provisioned ports the high-order cross-connection-rate (either the SDH-rate or its equivalent SONET-rate) must match with the high-order tributary-rate.

As an example a tributary provisioned to vc44c can be cross-connected only at a rate of vc44c (or STS12, which is equivalent) - requests with other rates will be rejected. This restriction concerning the consistency of tributary-rate and cross-connection-rate does NOT hold in the case the port is provisioned as adaptive (which is only possible in case of a SONET-port).

Low-order tributary-rate provisioning versus cross-connection-rate

For SONET low-order tributaries, only one type is allowed (VT1.5) which causes to match its rate with the referring cross-connection rate.

For SDH low-order tributaries, two types are allowed (LO-VC3 and VC12). VC12 tributaries are created by default. However, if a cross-connection request allows to adapt the signal structure to the requested rate then this request will be granted.

As an example, a high-order VC4 tributary has been substructured to 63 times VC12 signals (e.g. because a first VC12 cross-connection has been previously established for the first timeslot out of the 63). A cross-connection request to cross-connect the last third of the VC4 as a low-order cross-connection at a LO-VC3 will be granted since an adaptation is possible (no VC12 cross-connection present yet). Thus, the NE does an automatic correction of the substructure (in this case it changes to 42 times VC12 plus 1 times LO-VC3), before the cross-connect request will be successfully executed.

Please note that the following features are not part of the current 1675 LambdaUnite MSS release and are thus not specified here:

- provisioning of adjunct path-protected cross-connection (no adjunct related parameter)
- changing working and protection leg for path-protected cross-connections (no TL1 command OPR-WKGLEG)
- no adaptive-mode is allowed for both *in_aids* of a path-protected cross-connection.

AU3/AU4 Conversion Cross-Connections

AU3/AU4 conversion cross-connections allow the conversion of a HO STS1 (SONET) or VC3 (SDH) to a LO SDH VC3, and the mapping into a HO VC4.

From cross-connection provisioning point of view, an AU3/AU4 conversion cross-connection is a cross-connection between a HO STS1 or VC3 tributary and a LO VC3 tributary.

A conversion cross-connection has a HO or AU3 side and a LO or TU3 side.

In the requirements the terms “HO side” and “LO side” are used, to show the similarity of a conversion cross-connection with a HO cross-connection at one side and a LO cross-connection at the other side.

Contents

List of commands

Autonomous notification

TID Change

***loca*, *locz* auto notification for service traffic**

If a source TID (*loca*) or destination TID (*locz*) is changed due to a TID change of one NE within a BLSR ring/MS-SPRing, the update is reported by means of corresponding ED-CRS notifications.

No auto notification for *loca* and *locz* change for extra traffic

For extra traffic the *loca/z* info are generated automatically, for this reason they will change after an initialization or a recovery. Too many intermediate, void changes would be detected and notified.

No notification for *sqfa* and *sqfz* change

If a source TID (*loca*) or destination TID (*locz*) is changed due to a TID change of one NE within a MS-SPRing with the *protnmode* = TOPEX, the related parameters *sqfa* and *sqfz* will change to AINVALID respectively to MINVALID for an intermediate time, depending on internal processing speed and fault conditions of the ring. Finally they will again change to other values. Due to this transient behaviour it is only suitable to make it retrievable on demand.

No auto notification for loca and locz change caused by Automatic Squelch Map Calculation

If Automatic Squelch Map Calculation is used in a BLSR/MS-SPRING, then changes of *loca*/*locz* and the related status information *locastate*/*loczstate* caused by the Automatic Squelch Map Calculation shall not be notified. *loca*/*locz* and the related status information *locastate*/*loczstate* are generated automatically. For this reason they will change after an initialisation or a recovery. Too many intermediate, void changes would be detected and notified.

ONNS triggered changes

The ONNS TL1 command ENT-PATH shall lead to ENT-CRS notifications in the NEs along the route of the new path. In case of a 1+1 protected ONNS path there are also ENT-PROTN-GRP notifications in the source and destination NE of the ONNS path.

The ONNS TL1 command DLT-PATH shall lead to DLT-CRS notifications in the NEs along the route of the deleted path. In case of a 1+1 protected path there are also DLT-PROTN-GRP notifications in the source and destination NE of the ONNS path.

The ONNS TL1 command ED-PATH-ROLL

- shall lead to ENT-CRS and ENT-PROTN-GRP notifications in the two NEs where the bridge/merge function is performed
- shall lead to ENT-CRS notifications in the NEs along the new route of the path

- shall lead to DLT-CRS notifications in the NEs along the old route of the path
- shall lead to DLT-CRS and DLT-PROTN-GRP notifications in the two NEs where the bridge/selector is removed. The ONNS TL1 command ED-PATH-PROT
- shall lead to ENT-CRS notifications along the route of the new protecting path and ENT-PROTN-GRP notifications in the A and Z node when a modification from unprotected to 1+1 protected is performed
- shall lead to DLT-CRS notifications along the route of the new protecting path and DLT-PROTN-GRP notifications in the A and Z node when a modification from 1+1 protected to unprotected is performed
- shall not lead to additional host notifications when a modification from unprotected to auto-reroute or vice versa is performed.

The (non-revertive) restoration of an ONNS path shall lead to ENT-CRS notifications in the NEs along the restored route of the path, and to DLT-CRS notifications in the NEs along the route of the failed path.

The revertive restoration of an ONNS path shall lead to ENT-CRS notifications in the NEs along the restored route of the path, and to ENT-PROTN-GRP notifications in the A and Z node of the path. When the restoration reverts, it shall lead to DLT-CRS notifications in the NEs along the restored route of the path, and to DLT-PROTN-GRP notifications in the A and Z node of the path.

Tributary

Overview

Purpose

This section lists messages related to cross-connections on tributary level.

Contents

List of commands

Port

Overview

Purpose

This section contains messages related to cross-connections on port level.

Contents

List of commands

UserByte port

Overview

Purpose

Physically the UserByte-Port is a physical transmission interface, consisting of both an input and an output, which can be used to carry traffic through channels given by the E1- or F1-Byte (located in the RSOH) or the E2-Byte (located in the MSOH). Logically an UserByte-Port is an entity within an NE which models the function of the physical UserByte-Port.

Six UserByte-Ports are available (with AIDs denoted ubio1 .. ubio6) each of which can be associated to an optical port of an OPn either to the E1- or E2- or F1-Byte. The association is always bidirectional. Four of these six UserByte-Ports (AIDs: ubio1 .. ubio4) are provisionable to either the G703- or V11-interface, while the others (AIDs: ubio5,ubio6) are fixed to V11-interface.

The association as well as this mode setting can be done via the SET-OW-command. Furthermore the SET-OW command allows to release the association (dis-association). With the RTRV-OW command the information related to an UserByte-Port can be retrieved from the NE.

Note: Throughout this section the term order-wire (OW) is retained in the command names for ″historical″ reasons although the name user-byte would be more appropriate, because these ports can not only be used for order-wire purposes.

Contents

List of commands

Autonomous notification

Start-up with default database

The event *Start-up with default DB* causes the 6 instances of type UserByte port to be created.

Virtual switch

Overview

Purpose

Starting with the entity Virtual Switch the TL1 commands for all entities of the GE1 functionality will be described in the following chapters.

These are in detail:

- Virtual switch (VRTSW)
- Ethernet port (EPORT)
- Virtual concatenation group (VCG)
- Virtual concatenation tributary (VCGTRIB).

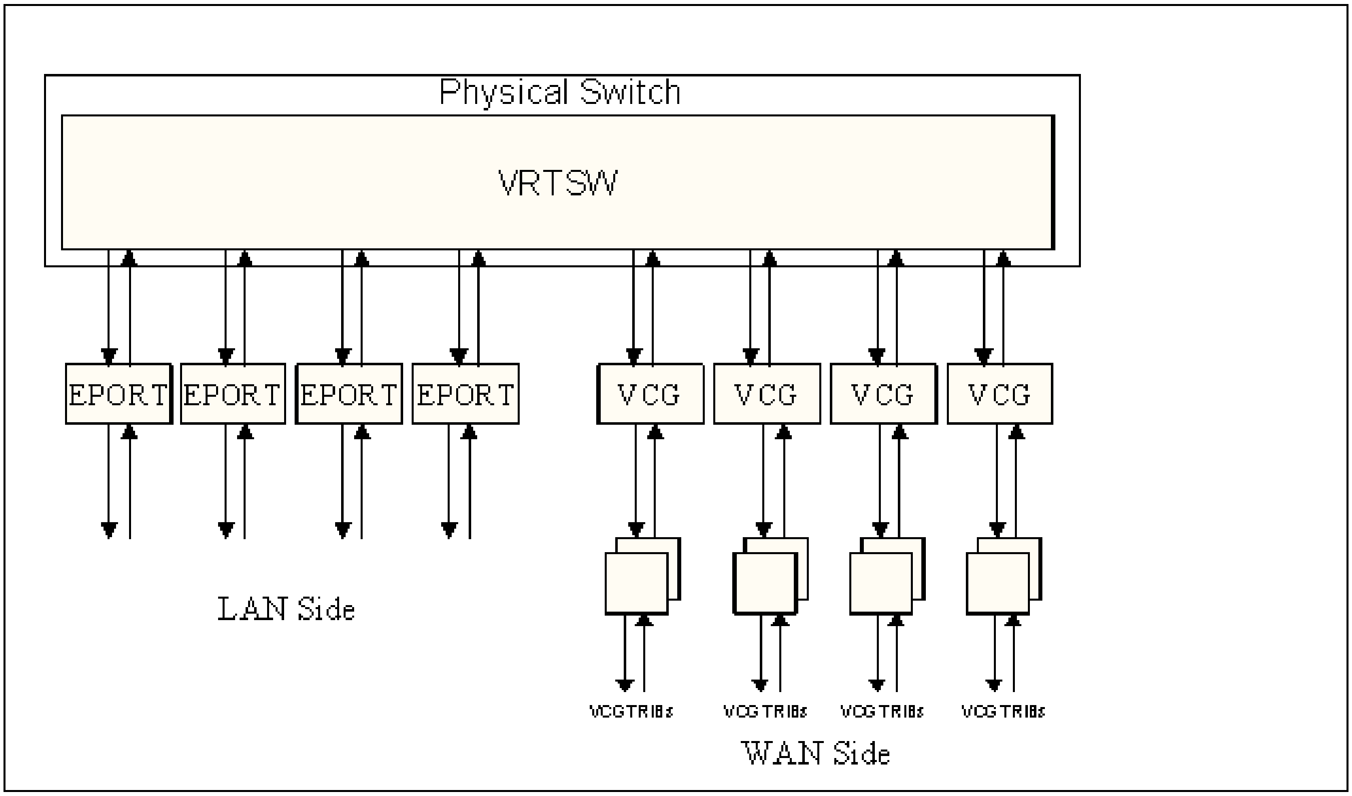
Each GE1 circuit pack consists of the following entities as shown in the following figure. The number of EPORT and VCG instances connected to the virtual switch is fixed. The number of VCGTRIBs per VCG depends on whether operating in SDH or SONET mode. In SDH mode there are 7 VCGTRIBs of type VC4 whereas in SONET mode there are 21 VCGTRIBs of type STS1s.

Commands for the entity *Virtual switch* allow to retrieve and modify its parameters. These commands are applicable to the GE1 circuit pack which supports one virtual switch instance.

The virtual switch as part of the physical switch on the GE1 pack is created by software. It serves to group and separate traffic within the physical switch via VLAN tagging.

The virtual switch is not applicable on the GE10PL1 pack.

The virtual switch is not applicable on the ES64U1 pack.


Physical Switch
VRTSW
EPORT
EPORT
EPORT
EPORT
VCG
VCG
VCG
VCG
LAN Side
VCGTRIBs
VCGTRIBs
VCGTRIBs
VCGTRIBs
WAN Side

Contents

List of commands

Ethernet port

Overview

Purpose

Starting with the entity Ethernet port the TL1 commands for all entities of the GE1 functionality will be described in the following chapters. These are in detail:

- Ethernet port (EPORT)
- Virtual concatenation group (VCG)
- Virtual concatenation group tributary (VCGTRIB).

GE1 circuit pack

Commands for the entity Ethernet port cover the parameters for

- physical link layer (layer 1) and
- data link layer (layer 2)

of the LAN side of Gigabit Ethernet. These commands are applicable to the GE1 circuit pack which has 4 LAN ports.

Those ethernet ports are mainly intended to connect several parts of a LAN via standard transmission technique like SDH or SONET.

There is no interworking with the Data Communications Network application (which uses Ethernet-LAN ports, too) used to control the NEs. The latter one is specified in another section.

GE10PL1 pack

Modelling of the transmission path

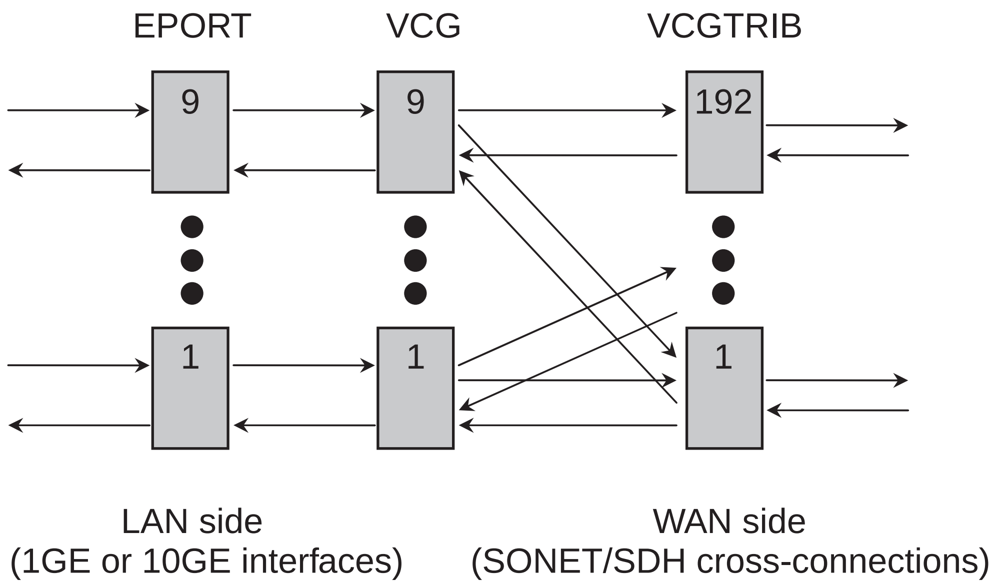
On the GE10PL1 pack, transmission functionality will be described by the following entities:

- Ethernet Port (EPORT)
- Virtual Concatenation Group (VCG)
- Virtual Concatenation Tributary (VCGTRIB).

Transmission entities on the GE10PL1/1A8

The GE10PL1/1A8 circuit pack consists of:

- 9 EPORTs (8 as 1 GE interface, 1 as 10GE interface)
- 9 VCGs (1 per EPORT)
- 192 VCGTRIBs (shared among the VCGs).



Existence of EPORT entities

EPORT entities exist only if the referring pluggable module has been successfully pre- or auto-provisioned. The first 8 ports are reserved for 1GE interface types while the 9th port is reserved for 10GE interface types. There will be only one transmission rate being accepted at the same time (either 8 times 1GE ports or 1 times 10GE port). Each entity represents functionality always bidirectionally.

Existence of VCG entities

VCG entities exist only if the referring pluggable module has been successfully pre- or auto-provisioned and are therefore created in conjunction to the referring EPORT entity. The same constraints are applicable here. Each entity represents functionality always bidirectionally.

Existence of VCGTRIB entities

VCGTRIB entities exist only if the referring pluggable module has been successfully pre- or auto-provisioned and are therefore created in conjunction to the referring VCG entity.

The number and address of the entities depend on the VCG type as follows:

VCG No. X exists	Existing VCGTRIB Numbers
No	None
Yes, for X=1…8	Yes, with 1…21
Yes, for X=9	Yes, with 1…192

Significance of VCGTRIB entities (general)

Out of the created VCGTRIB entities for a specific VCG, not all of them may be significant (configurations have no effect). This depends on the configured transmission rate and on the membership configuration.

The following table illustrates those dependencies:

Signal structure	VCG membership	VCGTRIB significance
211 (21 times STS1) or 1921 (192 times STS1)	Enable	Significant
	Disable	Insignificant
73 (7 times STS3C) or 74 (7 times VC4) or 644 (64 times CV4)	Enable	Significant (each third VCGTRIB only) [if (VCGTRIB No. 1) modulo 3 == 0)] Insignificant otherwise
	Disable	Insignificant

Significance of 10GE VCGTRIB entities (contiguous concatenation mode)

For the 10GE VCG, it is possible to configure a signal mapping into a contiguous concatenated signal instead of a mapping into a number of virtual concatenated signals. The mapping is controlled by the VCG parameter *vcat*. In this case, the first VCGTRIB entity remains significant while the other 191 entities will become insignificant (if not yet insignificant before). Note that the parameters of the VCG and VCGTRIB entities which are not applicable in the contiguous concatenation mapping mode will be set to default and cannot be altered as long this mode is present.

Meaning of the terms “Existing” and “Significance”

If an entity does not exist then any operation for this instance will be rejected. If an entity is insignificant then all allowed operation may be executed. Configured values will be stored in the database and can be retrieved. However, the settings have no impact to the transmission flow in the hardware.

Contents

List of commands

Autonomous notifications

GE1 circuit pack is auto-provisioned

If a GE1 circuit pack is auto-provisioned a GE1PORT instance will be created with the given default parameter values for all GE1 ports of the circuit pack.

This change is reported via the ENT-EPORT notification.

Optical Module for a GE10PL1/1A8 Circuit Pack is Auto-provisioned

If an optical module is successfully auto-provisioned as a socket of the GE10PL1/1A8 circuit pack then a related EPORT instance will be created with the specified default parameter values. In case that the socket with the Simple_SocketAID = sc9 has been created, the EPORT shall reflect an instance with a parameter value modifier = 10GEPL, otherwise with a parameter value modifier = 1GEPL. All applicable parameters listed in the ENT-EPORT notification shall be reported in a referring single ENT-EPORT notification.

Virtual concatenation group

Overview

Purpose

The Virtual Concatenation Group is an entity which exists for Gigabit Ethernet and Fully Transparent applications. For both applications the parameter set is slightly different, making a distinction of the related commands via a modifier for the Fully Transparent application necessary (e.g no LCAS is supported for the fully transparent application).

Overview of GE1 Application

Commands for the entity virtual concatenation group (VCG) cover the parameters for both

- the Ethernet side (WAN ports) and
- the EoS (Ethernet over SDH/SONET).

These commands are applicable to the GE1 circuit pack which has 4 WAN ports.

Note: The virtual concatenation group (VCG) and its tributaries (VCGTRIB) have some direction dependent parameters. Following the semantic for optical ports the direction indicators input and output are used.

Input direction

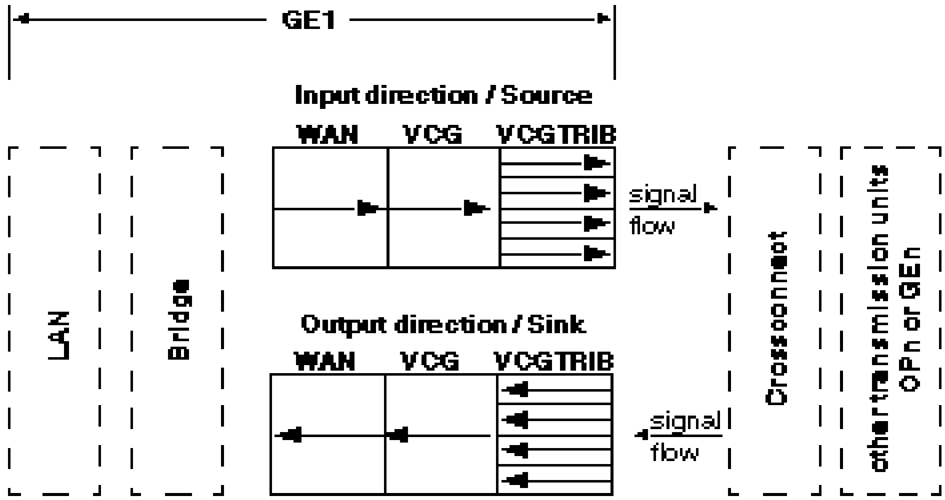
WAN → VCG → VCGTRIB → Cross-connect

As those tributaries have termination function, this is also often called source.

Output direction

Cross-connect → VCGTRIB → VCG → WAN

As those tributaries have termination function, this is also often called sink.



Overview of OPT*n* Application

Commands for the entity virtual concatenation group (VCG) cover the parameters for both

- the ODU layer and
- the Virtual Concatenation functionality.

These commands are applicable to the OPT2G5 circuit pack which has 3 client ports (one-to-one relationship to the VCG ports).

Note: The virtual concatenation group (VCG) and its tributaries (VCGTRIB) have some direction dependent parameters. Following the semantic for optical ports the direction indicators input and output are used. Also the equivalent terms ingress and egress are applicable.

Input direction

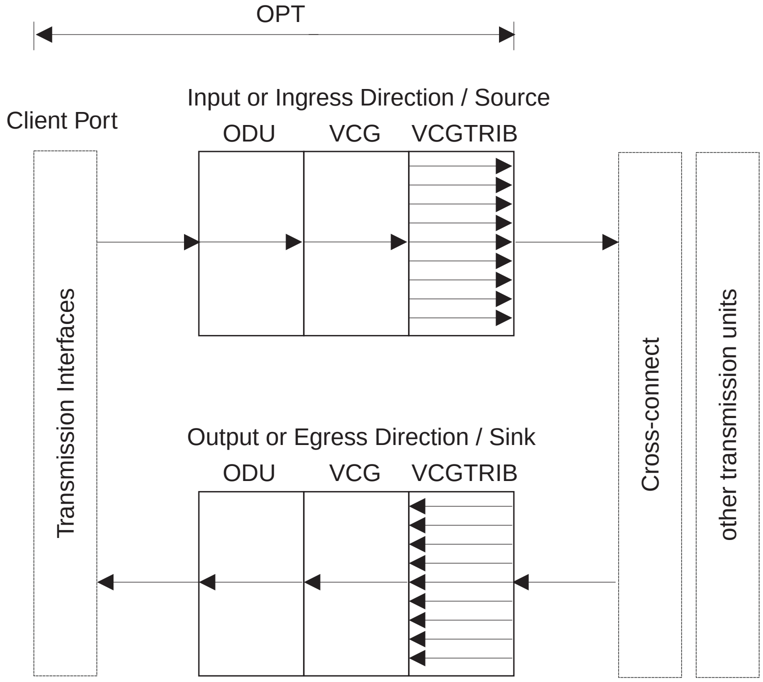
Client Port → VCG (including ODU) → VCGTRIB → Cross-connect

As those tributaries have termination functions, this is also often called source.

Output direction

Cross-connect → VCGTRIB → VCG (including ODU) → Client Port

As those tributaries have termination functions, this is also often called sink.



Overview of ES64U application

The ES64U has no LAN ports on the pack.

The virtual switch is other than for the GE1 application not managed via TL1 for the ES64U application. For the ES64U application it is managed via a direct CLI/SNMP interface to the data part of the ES64U.

The WAN port setting is other than for the GE1 application not managed via TL1 for the ES64U application. For the ES64U application it is managed via a direct CLI/SNMP interface to the data part of the ES64U.

Commands for the entity virtual concatenation group (VCG) cover the parameters for EoS (Ethernet over SDH/SONET).

These commands are applicable to the ES64U circuit pack which has up to 252 VCG ports.

Note: The virtual concatenation group (VCG) and its tributaries (VCGTRIB) have some direction dependent parameters. Following the semantic for optical ports the direction indicators input and output are used.

Input direction

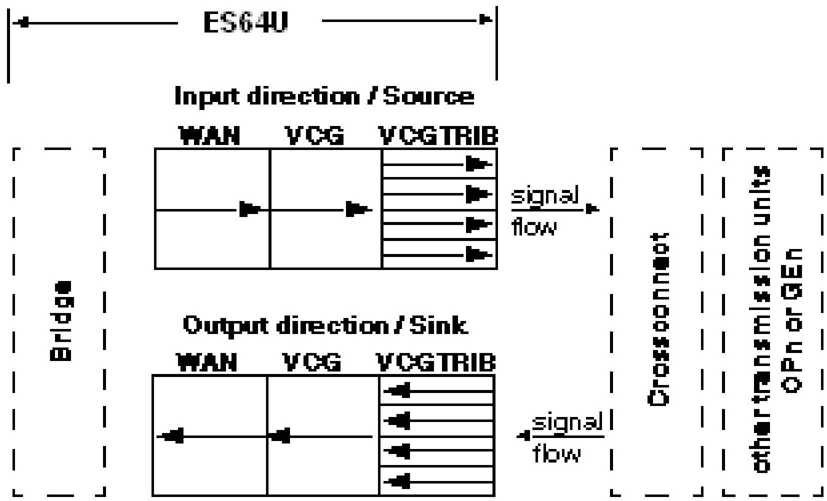
WAN → VCG → VCGTRIB → Cross-connect

As those tributaries have termination function, this is also often called source.

Output direction

Cross-connect → VCGTRIB → VCG → WAN

As those tributaries have termination function, this is also often called sink.



From the Ethernet bridge point of view the definition of input and output direction is quite confusing as it is contrary to expectation. However, the definition was made such that it fits to the usual nomenclature being used for optical ports from the cross-connect point of view. The terms input and output direction are e.g. used for parameters inputsig, istd, and unequiposig which are available on standard OC***n***/STM***n*** ports also.

For an overview about the resource allocation for an ES64U in a slot, the command **RTRV-RESALLOC-VCG** can be used. This command is needed to provide the user an overview about the backplane fragmentation and where the VCGs are located.

Contents

List of commands

Autonomous notifications

GE1 circuit pack is auto-provisioned

If a GE1 circuit pack is auto-provisioned a VCG instance will be created with the given default parameter values for all VCGs of the circuit pack.

This change is reported via the **`ENT-VCG`** notification.

OPT*n* Circuit Pack is Auto-provisioned

If an OPT***n*** circuit pack is auto-provisioned a VCG instance will be created with the given default parameter values for all Virtual Concatenation Groups of the circuit pack. This change shall be reported via the **`ENT-VCG-TRANS`** notification.

Virtual concatenation group tributary

Overview

Purpose

For an overview of the GE1 related entities EPORT, VCG, and VCGTRIB see "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547).

For an overview of the OPT***n*** related entities EPORT, VCG, and VCGTRIB (and client port), see "Overview of OPTn Application" (p. 7-25).

For an overview of the GE10PL1 related entities EPORT, VCG, and VCGTRIB, see "GE10PL1 pack" (p. 7-19).

For an overview of the ES64U related entities VCG and VCGTRIB, see "Overview of ES64U application" (p. 7-26).

Commands for the entity virtual concatenation group tributary cover the parameters for

- standard termination as for other tributaries
- tributary related parameters of virtual concatenation.

Each virtual concatenation tributary can be configured independently.

Contents

List of commands

Autonomous notifications

GE1 circuit pack is auto-provisioned

If a GE1 circuit pack is auto-provisioned 7 VC4 or 21 STS1 VCGTRIB instances will be created (dependent on the attribute *istddflt*) with the given default parameter values for all VCGs of the circuit pack.

This change is reported via the **ENT-VCGTRIB** notification.

OPT*n* circuit pack is auto-provisioned

If an OPT***n*** circuit pack is auto-provisioned 17 VC4 or STS3c VCGTRIB instances will be created (dependent on the attribute **istddflt**) with the given default parameter values for all Virtual Concatenation Groups of the circuit pack.

This change shall be reported via the **ENT-VCGTRIB** notification.

Optical module for a GE10PL1/1A8 circuit pack is auto-provisioned

If an optical module is successfully auto-provisioned as a 1 GE socket of the GE10PL1/1A8 circuit pack then 21 VCGTRIB instances will be created with the specified default parameter values.

If an optical module is successfully auto-provisioned as a 10 GE socket of the GE10PL1/1A8 circuit pack then 192 VCGTRIB instances will be created with the specified default parameter values.

All applicable parameters listed in the **ENT-VCGTRIB** notification shall be reported in a referring single **ENT-VCGTRIB** notification.

VCG on a ES64U circuit pack is provisioned

If a VCG is successfully provisioned on a ES64U circuit pack, then the VCGTRIB instances belonging to it will be created with the specified default parameter values. The VCGTRIB number range is from 1 up to the number of tributaries given by the parameter *inputsig* given with the **ENT-VCG** command. All applicable parameters listed in the **ENT-VCGTRIB** notification shall be reported in a referring single **ENT-VCGTRIB** notification.

VCG on a ES64U circuit pack is deprovisioned

If a VCG is successfully deprovisioned on a ES64U circuit pack, then the VCGTRIB instances belonging to it will be deleted. All applicable parameters listed in the **ENT-VCGTRIB** notification shall be reported in a referring single **ENT-VCGTRIB** notification.

Group notifications for VCG tributaries

For efficiency reasons the notifications for VCG tributaries shall be sent out as a single notification (so-called group notification) using a range over the aid which depends on the used mapping of the associated vcg port, i.e. the Simple_VCGNumAID field in the VCGNumAID is set to

- 1&&-<N> for sequential range
- 1&-4 .. &-<N> for non-sequential range

with N is maximal legal VCG tributary number.

Example:

The AID field in a group notification for VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping)

1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping)

For efficiency reasons the notifications for tributaries shall be sent out as a single notification (so-called group notification) using a range over the aid which depends on the port rate of the associated port, i.e. the Simple_STS1NumAID field in the STS1NumAID is set to

- 1&&-<N>

with N is maximal legal STS-1/AU-3 number.

Example:

The AID field in a group notification for tributaries related to port 1 for a OP10 pack in slot 1 looks like:

1-1-#-#-1-1-1&&-192

Behavior

The **`ENT-VCG`** shall generate a **`REPT DBCHG`** message for the creation of the VCG together with a **`REPT DBCHG`** message for the creation of the VCG tributaries (**`ENT-VCGTRIB`** notification).

The command shall establish a virtual concatenation group together with the virtual concatenation group tributaries belonging to it according to the parameters given with this command. The virtual concatenation group tributary number range is from 1 up to the number of tributaries given by the parameter *inputsig*. The modification shall be reported by means of an **`ENT-VCG`** notification and grouped notifications for VCG tributaries.

Group notification for VCG tributaries

For efficiency reasons the notifications for VCG tributaries shall be sent out as a single notification (so-called group notification) using a range over the aid which depends on the used mapping of the associated VCG port, i.e. the *Simple_VCGNumAID* field in the *VCGNumAID* is set to:

- 1&&-***<N>*** for sequential range
- 1&-4 … &-***<N>*** for non-sequential range

where ***<N>*** is the maximal legal VCG tributary number.

Example:

The AID field in a group notification for a VCG tributaries related to VCG port 1 for a GE1 pack in slot 1 looks like:

1-1-#-#-1-v1-1&&-21 (for STS-1/VC-3 mapping)

1-1-#-#-1-v1-1&-4&-7&-10&-13&-16&-19 (for STS-3c/VC-4 mapping)

Incoming VCG tributary is dropped or added

If an incoming (sink direction) VCG tributary is dropped or added due to manual operation or a failure event, the parameter *outmbst* shall be set to the corresponding value considering the current LCAS mode. This change shall be reported via the ED-VCGTRIB notification with the parameter *outmbst*.

For the modifiers VCT or STST tributaries will be added or dropped only via manual operation.

Outgoing VCG tributary is dropped or added

If an outgoing (source direction) VCG tributary is dropped or added due to manual operation or a failure event, the parameter *inmbst* shall be set to the corresponding value considering the current LCAS mode. This change shall be reported via the ED-VCGTRIB notification with the parameter *inmbst*.

For the modifiers VCT or STST tributaries will be added or dropped only via manual operation.

Lower-order tributaries

Overview

Purpose

Higher-order tributaries on I/O packs (e.g. OPn, EP155) can be substructured as follows:

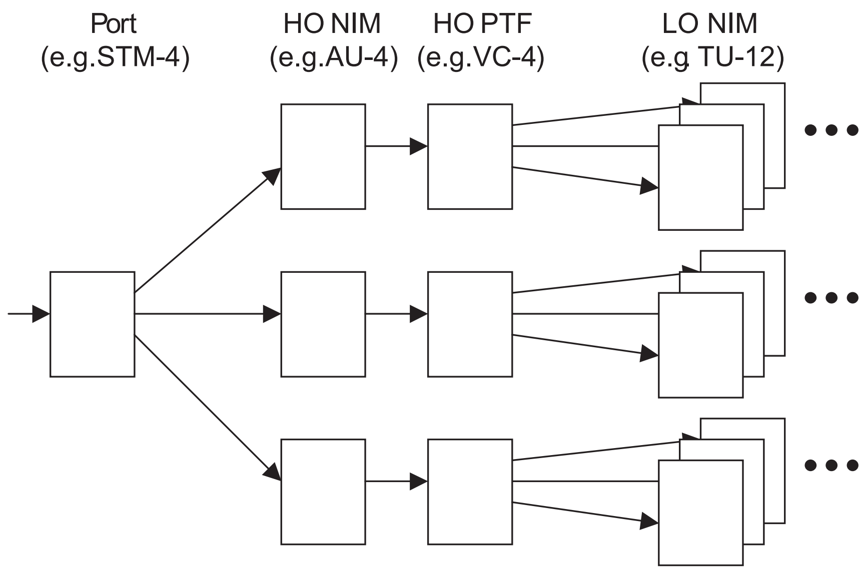
- HO path termination function (PTF, see SRD_TL1_Configuration)
- LO non-intrusive monitoring function (see below).

Following substructures are supported:

- SDH: VC4 PTF entity and 3 VC3 or 63 VC12 LONIM entities - or a mix of it,
- SONET: STS1 PTF entity and 28 VT1.5 LONIM entities.

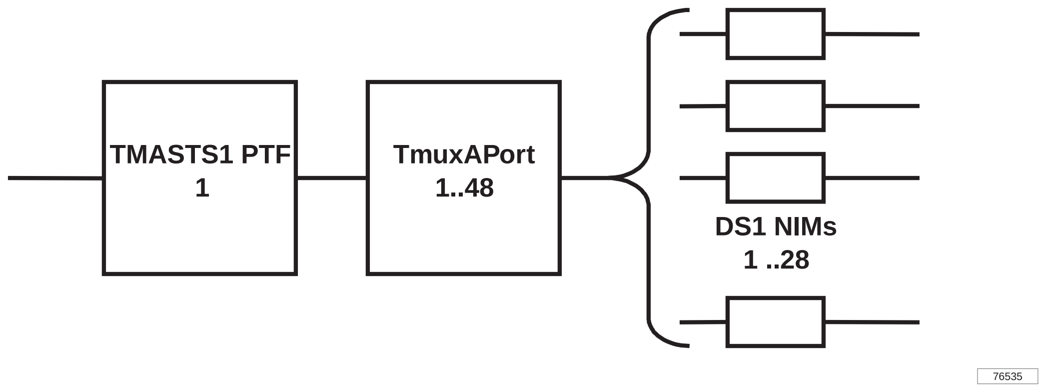
Example

The following figure shows an example:



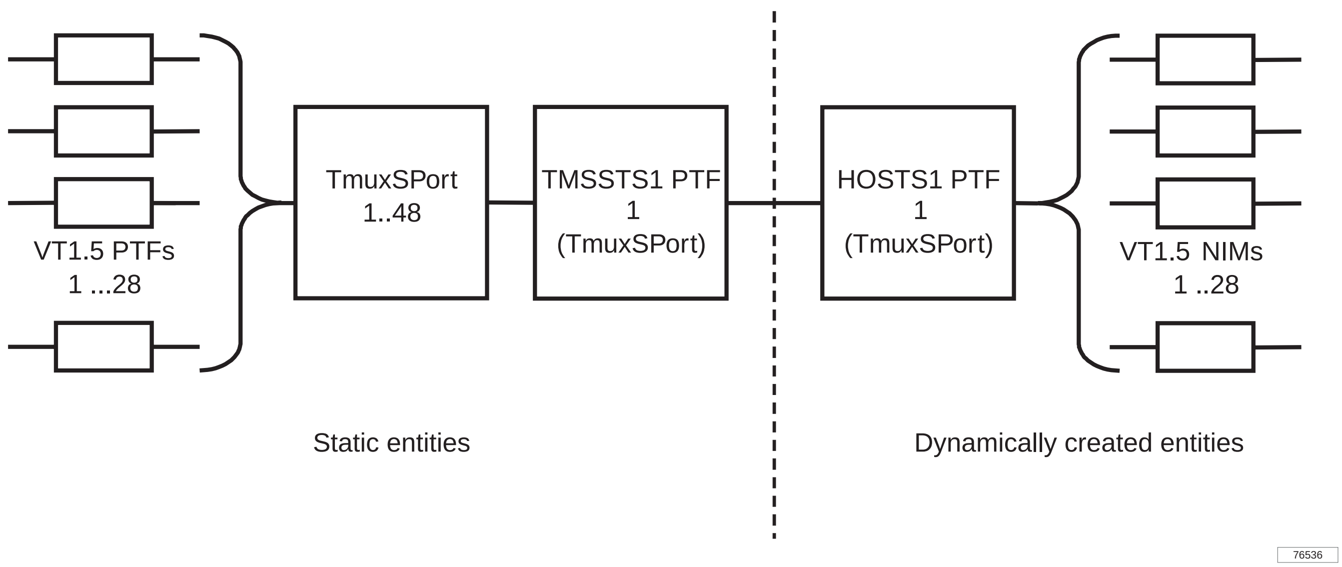
HO and LO entities on the asynchronous side of the TransMUX pack

The entities at the asynchronous side of the TransMUX pack are static, i.e. are created when the TransMUX capacity is set away from 0 and persist until the TransMUX capacity is set to 0 again.



HO and LO entities on the synchronous side of the TransMUX pack

The static entities at the synchronous side of the TransMUX pack are created when the TransMUX capacity is set away from 0 and persist until the TransMUX capacity is set to 0 again. The dynamically created entities are created with the first LO cross-connection at the STS1 PTF and removed when the last LO cross-connection at the PTF is deleted



Contents

List of commands

Remote Test Access

Overview

Purpose

Test access consists of a set of non-intrusive operations (“monitoring” modes) and intrusive operations (“split modes”) that aid users in turning up facilities and in fault sectionalization of existing service circuits.

Basic user capabilities in test access include:

- assigning available tributary(ies) as “test access tributary(ies)”, whose line interface(s) will be connected by the user to the external testing system. These test access tributaries are assigned and released as part of establishing or disconnecting test access sessions.
- establishing test access cross-connections from tributaries to be tested to the test access tributary(ies) that either monitor or “split” connections for testing purposes.
- retrieving information about test access connections present on the system.
- releasing test access connections and test tributaries, upon which the system automatically restores any pre-existing cross-connections that were affected.

The architecture and nomenclature depicted below help to explain how test access is defined and used.

The E-Tributary and F-Tributary are the tributaries unter test. The E and F tributaries are the ends of a one-way or two-way cross-connection, with the dashed lines indicating the signal traveling through the system. The signal path from the E-end to the F-end is referred to as the ″A″ direction; the opposite direction as the ″B″ direction. If there is only a one-way cross-connection, the signal travels in the A direction. If an idle (un-cross-connected) tributary is the focus of test access, it is always the E-end.

In the context of Remote Test Access a Higher-Order (HO) connection topology is considered one-way if the Reported Leg Pair of the cross-connection between the involved tributaries indicates 1WAY (repleg = 1WAY). In the context of Remote Test Access a Higher-Order (HO) connection topology is considered two-way if the Reported Leg Pair of the cross-connection between the involved tributaries indicates 2WAY (repleg = 2WAY).

In the context of Remote Test Access a Lower-Order (LO) connection topology is considered one-way if there is a cross-connection from tributary #1 to tributary #2 AND there is no cross-connection in opposite direction between tributaries #1 and #2

This is applicable regardless whether the connection from #1 to #2 is unprotected, bridged or path-protected. Also note, that there might be other cross-connections from any other tributary with tributary #1 as output tributary.

In the context of Remote Test Access a Lower-Order (LO) connection topology is considered two-way if there is a cross-connection from tributary #1 to tributary #2 AND there is also a cross-connection in opposite direction from tributary #2 to tributary #1 This is applicable regardless whether the connections between tributaries #1 and #2 are unprotected, bridged or path-protected or any combination of both.

The following table shows the supported test access modes.

Two groups of test access modes are defined:

- single test access tributary modes
- dual test access tributary modes.

Mode	Description	Group
MONE	Monitor E-end mode of test access. This test access mode monitors the E end of an idle tributary or a leg of a cross-connection.	Single
MONF	Monitor F-end mode of test access. This test access mode monitors the F-end of a two-way cross-connection.	Single
MONEF	Monitor E-end and F-end mode of test access. This test access mode monitors (E end and F end) both directions of a 2WAY cross-connection.	Dual
SPLTE	Split access E-end mode of test access. This test access mode splits the connection between the E end and F end of a 1WAY cross-connection or a leg of 2WAY cross-connection. Can also be used in an idle tributary.	Single
SPLTF	Split access F-end mode of test access. This test access mode splits the E-end and F-end connection of a two-way cross-connection to allow testing at the F-end.	Single
SPLTEF	Split E-end and F-end mode of test access. This test access mode splits both directions of a 2WAY cross-connection to allow testing at both the E end and F end.	Dual
SPLTA	Split access “A” direction mode of test access. This test access mode splits the connection between the E and F tributaries of a one-way or two-way cross-connection to allow testing in the “A” direction.	Single

Mode	Description	Group
SPLTB	Split access “B” direction mode of test access. This test access mode splits the E-end and F-end connection of a two-way cross-connection to allow testing in the “B” direction.	Single
SPLTAB	Split access A and B directions mode of test access. This test access mode splits both directions of a two-way cross-connection to allow testing in both the A and B directions.	Dual
TERME	Terminated test access mode. This test access mode splits the signal, sending the input to the testing system and injecting AU-AIS/AIS-P toward the path selector to allow monitoring the incoming signal of the leg that is not currently carrying service traffic without the risk that the path selector will switch away from the service-carrying leg.	Single

Topology in Remove Cross-connection Command (TL1-parameter: *cct*)	Topology of Existing Atomic Cross-connection (internal parameter: ″Atomic cross-connection topology″ as described in the RTRV-CRS section in the *repleg* context)	Leg Status of Existing Atomic Cross-connection (internal parameter: ″Leg status″ as described in the RTRV-CRS section in the repleg context)	Validation Response (″Accept″ means that the command is replied by a COMPLD, ″Deny″ means, that the command is rejected with the Error Code as stated above)
1WAY	point-to-point	1-way	Accept
1WAY	point-to-point	2-way	Accept
1WAY	pathprotected	1-way	Deny
2WAY	point-to-point	1-way	Deny
2WAY	point-to-point	2-way	Accept
2WAY	pathprotected	1-way	Deny
1WAYPPROT	point-to-point	1-way	Deny
1WAYPPROT	point-to-point	2-way	Deny
1WAYPPROT	pathprotected	1-way	Accept

Contents

List of commands

ONNS to UPSR/SNCP Constructs (OUC)

Overview

Purpose

This section lists the TL1 commands related to the UPSR/SNCP constructs.

Contents

List of commands

8 TL1 messages - protection

Overview

Purpose

This chapter contains the TL1 interface definition for the domain "protection".

Contents

Protection group

Overview

Purpose

The protection group commands cover all types of protections within the system. The following protection group types can be distinguished:

- 1+1 equipment protection group
- 1+1 multiplex section/line protection group
- 2-fibre MS-SPRing/BLSR protection group
- 4-fibre MS-SPRing/BLSR protection group
- Path protection group.

Dedicated subsections corresponding to these protection group types will cover the specific behavior. If such a separation is not made, the description is valid for all protection group types. In addition common properties are also described in a dedicated General section.

Three operation modes can be distinguished for a 4-fibre MS-SPRing/BLSR protection group:

1. 2-sided node: An east side (protection and worker) and a west side (protection and worker) is provisioned for that group. Therefore, four ports are members within that protection group.
2. 1-sided node: Only one side is provisioned for the protection group. That can be either the east or the west side. Therefore, two ports are members within that protection group. This mode is used in context of an open 4-fiber ring w.r.t. end node in that chain.
3. Asymmetric node: Both protection ports are members of the protection group and only one or zero worker ports are assigned externally.

All attributes/parameters specifying east/or west side properties are optional and will be omitted if the corresponding side is not provisioned in case of a 1-sided node.

Note: 1-sided node (open ring) is not supported in R1. As indication if and when this feature is supported use the following criteria (supported if criteria is met): Ports can be added/removed to/from one side of the protection group (conversion from 1-side node to 2-side node and the other way round). Be aware to differentiate that scenario from switching from an asymmetric node to a non-asymmetric and the other way round. The combination of a 1-sided and asymmetric node is not allowed.

Contents

Introduction to protection groups

General information

The following section presents on overview of all attributes of the different protection types and their use within the different commands. The information is derived from the corresponding sections describing the commands and notifications.

Note: The following tables are specifying the set of operations, notifications, and attributes for protection groups. They do not consider any release information. Check in the corresponding sections in which release the functionality is supported.

1+1 equipment protection group

Entity table for switching and controller equipment protection group (EQPTSWFBR and EQPTCTL)

Attribute	Opr	Not	Class	Note
aid – Switch And Timing Protection Group AID or Controller Protection Group AID	RTRV	ENT DLT PROTNSW	input	(1)
protype – Protection Type	RTRV	ENT DLT PROTNSW	input	
wkg – Worker Unit	RTRV	ENT PROTNSW	input	(2)
protn – Protection Unit	RTRV	ENT PROTNSW	input	(2)
actunit – Active Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl – Worker Fault Status	RTRV		state	

Note 1: protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Entity table for EP155 Equipment Protection Group (EQPTEP155)

Attribute	Opr	Not	Class	Note
aid – EP155 equipment protection group AID	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	(1)

Attribute	Opr	Not	Class	Note
protype – Protection Type	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	
wkg – Worker Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
protn – Protection Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
actunit – Active Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl – Worker Fault Status	RTRV		state	

Note 1: protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Entity table for EP51 Equipment Protection Group (EQPTEP51)

Attribute	Opr	Not	Class	Note
aid – EP51 Equipment Protection Group AID	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	(1)
protype – Protection Type	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	
wkg – Worker Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
protn – Protection Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
actunit – Active Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl – Worker Fault Status	RTRV		state	

Note 1: protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Entity table for LOXC Equipment Protection Group (EQPTLOXC)

Attribute	Opr	Not	Class	Note
aid – LOXC Equipment Protection Group AID	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	(1)
protype – Protection Type	RTRV, ENT, DLT	ENT, DLT, PROTNSW	input	
wkg – Worker Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
protn – Protection Unit	RTRV, ENT	ENT, PROTNSW	input	(2)
actunit – Active Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl – Worker Fault Status	RTRV		state	

Note 1: protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

TransMUX equipment protection group

Entity table for TransMUX equipment protection group (EQPTTMUX)

Attribute	Opr	Not	Class	Note
aid - TransMUX Equipment Protection Group AID	RTRV	ENT	input	(1)
	ENT	DLT		
	DLT	PROTNSW		
protype - Protection Type	RTRV	ENT	input	
	ENT	DLT		
	DLT	PROTNSW		
wkg - Worker Unit	RTRV	ENT	input	(2)
	ENT	PROTNSW		
protn - Protection Unit	RTRV	ENT	input	(2)
	ENT	PROTNSW		
actunit - Active Unit State	RTRV	PROTNSW	state	(3)
swreq - Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl - Protection Fault Status	RTRV		state	

Attribute	Opr	Not	Class	Note
wkgsiglvl - Worker Fault Status	RTRV		state	

Note 1: Protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: Protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

ES64U equipment protection group

Entity table for ES64U equipment protection group (EQPTES64U)

Attribute	Opr	Not	Class	Note
aid - ES64U Equipment Protection Group AID	RTRV	ENT	input	(1)
	ENT	DLT		
	DLT	PROTNSW		
protype - Protection Type	RTRV	ENT	input	
	ENT	DLT		
	DLT	PROTNSW		
wkg - Worker Unit	RTRV	ENT	input	(2)
	ENT	PROTNSW		
protn - Protection Unit	RTRV	ENT	input	(2)
	ENT	PROTNSW		
actunit - Active Unit State	RTRV	PROTNSW	state	(3)
swreq - Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl - Protection Fault Status	RTRV		state	
wkgsiglvl - Worker Fault Status	RTRV		state	

Note 1: Protection switch events reported via **REPT EVT** message will use protection group aid for identification purpose.

Note 2: Protection switch events reported via **REPT SW** message will use worker/protection unit aid for identification purpose.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Multiplex section/line protection group

Entity table for 1+1 / 1:1 multiplex section/line protection group.

Attribute	Opr	Not	Class	Note
aid – protection group AID	RTRV ENT ED DLT	ENT ED DLT	input user	(1)
protype – Protection Type	RTRV ENT DLT	ENT DLT PROTNSW	input	
wkg – Worker Port	RTRV ENT ED DLT	ENT PROTNSW	input user	(2)
protn – Protection Port	RTRV ENT ED	ENT PROTNSW	input user	(2)
apspn – MSP-APS ASAP Pointer	RTRV ED	ENT ED	user	
rme – Protection Reversion	RTRV ENT	ENT	user	
wtr – Wait To Restore Time	RTRV ED	ENT ED	user	
protocol – APS Protocol	RTRV ENT	ENT	user	
dcclmsprotmd – DCC Line/MS Protection Mode	RTRV ED	ENT ED	user	
dccsrsprotmd – DCC Sect/RS Protection Mode	RTRV ED	ENT ED	user	
pgrate – protection group Rate	RTRV	PROTNSW	user	
actunit – Active Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl – Worker Fault Status	RTRV		state	
actunitfe – Far-End Incoming Requested Selection	RTRV		state	
actunitne – Near-End Outgoing Requested Selection	RTRV		state	
aps – Far-End Incoming APS	RTRV		state	
apsout – Near-End Outgoing APS	RTRV		state	

Note 1: AID field “transmission protection group ID” of protection group can be specified as part of **ENT-PROTN-GRP** command.

Note 2: Member port can be used as alias for identification of protection group entity.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Multiplex section/line protection group 1:N

Entity table for 1+N multiplex section/line protection group.

Attribute	Opr	Not	Class	Note
aid – protection group AID	RTRV ENT ED DLT	ENT ED DLT	input user	(1)
protype – Protection Type	RTRV ENT DLT	ENT DLT PROTNSW	input	
protn – Protection Port	RTRV ENT ED	ENT PROTNSW	input	(2)
apspn – MSP-APS ASAP Pointer	RTRV ED	ENT ED	user	
rme – Protection Reversion	RTRV ENT	ENT	user	
wtr – Wait To Restore Time	RTRV ED	ENT ED	user	
protocol – APS Protocol	RTRV ENT	ENT	user	
dcclmsprotmd – DCC Line/MS Protection Mode	RTRV ED	ENT ED	user	
dccsrsprotmd – DCC Sect/RS Protection Mode	RTRV ED	ENT ED	user	
pgrate – protection group Rate	RTRV	PROTNSW	user	
selunit – Selected Unit State	RTRV	PROTNSW	state	(3)
swreq – Switch Request State	RTRV	PROTNSW	state	(3)
protnsiglvl – Protection Fault Status	RTRV		state	
wkgsiglvl#***i*** (***i*** = 1…14) – Worker Channel #***i*** Fault Status	RTRV		state	
requnitfe – Far-End Incoming Requested Selection	RTRV		state	
requnitne – Near-End Outgoing Requested Selection	RTRV		state	
aps – Far-End Incoming APS	RTRV		state	
apsout – Near-End Outgoing APS	RTRV		state	

Note 1: AID field “transmission protection group ID” of protection group can be specified as part of **ENT-PROTN-GRP** command.

Note 2: Member port can be used as alias for identification of protection group entity.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

2-fibre BLSR/MS-SPRing protection group

Entity table for 2-fibre BLSR/MS-SPRing protection group.

Attribute	Opr	Not	Class	Note
aid – protection group AID	RTRV ENT ED DLT	ENT ED DLT	input user	(1)
protype – Protection Type	RTRV ENT ED DLT	ENT ED DLT PROTNSW	input	
rid – Ring Identification	RTRV ENT ED DLT	ENT ED	input user	(4)
east – East Port	RTRV ENT ED DLT	ENT PROTNSW	input user	(2)
west – West Port	RTRV ENT ED DLT	ENT PROTNSW	input user	(2)
blsrpn – MSSPRING-BLSR ASAP Pointer	RTRV ED	ENT ED	user	
wtr – Wait To Restore Time	RTRV ED	ENT ED	user	
pgrate – protection group Rate	RTRV	PROTNSW	user	
eswreq – East Switch Request State	RTRV	PROTNSW	state	(3)
wswreq – West Switch Request State	RTRV	PROTNSW	state	(3)
ringndst – Ring Node APS State	RTRV		state	
eapschst – Incoming East APS Channel Status	RTRV		state	
wapschst – Incoming West APS Channel Status	RTRV		state	
eapsdnid – Incoming East APS Destination Node Id	RTRV		state	
wapsdnid – Incoming West APS Destination Node Id	RTRV		state	
eapsprotn – Incoming East APS Protection Path	RTRV		state	
wapsprotn – Incoming West APS Protection Path	RTRV		state	
eapssnid – Incoming East APS Source Node Id	RTRV		state	
wapssnid – Incoming West APS Source Node Id	RTRV		state	
eapssw – Incoming East APS Switch Priority	RTRV		state	
wapssw – Incoming West APS Switch Priority	RTRV		state	
elprotnsw – East Last protection switch Attempt	RTRV		state	
wlprotnsw – West Last protection switch Attempt	RTRV		state	

Note 1: AID field “transmission protection group ID” of protection group can be specified as part of **ENT-PROTN-GRP** command.

Note 2: Member port can be used as alias for identification of protection group entity.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Note 4: Parameter is called *newrid* in the context of **ED-PROTN-GRP** command.

4-fibre BLSR/MS-SPRing protection group

Entity table for 4-fibre BLSR/MS-SPRing protection group.

Attribute	Opr	Not	Class	Note
aid – protection group AID	RTRV ENT ED DLT	ENT ED DLT	input user	(1)
protype – Protection Type	RTRV ENT ED DLT	ENT ED DLT PROTNSW	input	
rid – Ring Identification	RTRV ENT ED DLT	ENT ED	input user	(4)
eastwkg – East Worker Port	RTRV ENT ED	ENT ED PROTNSW	input user	(2)(5)
westwkg – West Worker Port	RTRV ENT ED	ENT ED PROTNSW	input user	(2)(5)
eastprotn – East Protection Port	RTRV ENT ED	ENT ED PROTNSW	input user	(2)(5)
westprotn – East Protection Port	RTRV ENT ED	ENT ED PROTNSW	input user	(2)(5)
blsrpn – MSSPRING-BLSR ASAP Pointer	RTRV ED	ENT ED	user	
wtr – Wait To Restore Time	RTRV ED	ENT ED	user	
swtr – Span Wait To Restore Time	RTRV ED	ENT ED	user	
protnmode – protection switch Mode	RTRV ENT ED	ENT ED	user	
pgrate – protection group Rate	RTRV	PROTNSW	user	
eswreq – East Switch Request State	RTRV	PROTNSW	state	(3)
wswreq – West Switch Request State	RTRV	PROTNSW	state	(3)
eactunit – East Active Unit State	RTRV	PROTNSW	state	(3)
wactunit – West Active Unit State	RTRV	PROTNSW	state	(3)
ringndst – Ring Node APS State	RTRV		state	
eapschst – Incoming East APS Channel Status	RTRV		state	

Attribute	Opr	Not	Class	Note
wapschst – Incoming West APS Channel Status	RTRV		state	
eapsdnid – Incoming East APS Destination Node Id	RTRV		state	
wapsdnid – Incoming West APS Destination Node Id	RTRV		state	
eapsprotn – Incoming East APS Protection Path	RTRV		state	
wapsprotn – Incoming West APS Protection Path	RTRV		state	
eapssnid – Incoming East APS Source Node Id	RTRV		state	
wapssnid – Incoming West APS Source Node Id	RTRV		state	
eapssw – Incoming East APS Switch Priority	RTRV		state	
wapssw – Incoming West APS Switch Priority	RTRV		state	
elprotnsw – East Last protection switch Attempt	RTRV		state	
wlprotnsw – West Last protection switch Attempt	RTRV		state	

Note 1: AID field “transmission protection group ID” of protection group can be specified as part of **ENT-PROTN-GRP** command.

Note 2: Member port can be used as alias for identification of protection group entity.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Note 4: Parameter is called *newrid* in the context of **ED-PROTN-GRP** command.

Note 5: Port can be used as user parameter in **ED-PROTN-GRP** in case of adding/removing one side to/from the protection group (conversion from 1-side node to 2-side node and the other way round). It can also be used to convert a 2-side node to an asymmetric node and the other way round.

Path protection group

Entity table for path protection group

Attribute	Opr	Not	Class	Note
aid – protection group AID	RTRV ED DLT	ENT ED DLT	input	

Attribute	Opr	Not	Class	Note
protype – Protection Type	RTRV ED DLT	ENT ED DLT	input	
ppgname – Path protection group Name	RTRV ENT ED	ENT ED	input user	(1)(2)(5)
wkg – Worker Tributary	RTRV ENT	ENT ED	user	(1)
protn – Protection Tributary	RTRV ENT	ENT ED	user	(1)
dripathpn – SNCP-UPSR ASAP Pointer	RTRV ED	ENT ED	user	
rme – Protection Reversion	RTRV ENT ED	ENT ED	user	(1)
wtr – Wait To Restore Time	RTRV ENT ED	ENT ED	user	(1)
htmr – Hold-Off Time	RTRV ENT ED	ENT ED	user	(1)
snctype – SNC Protection Type	RTRV ENT ED	ENT ED	user	(1)
outtribaid – Output Tributary AID	RTRV ENT	PROTNSW	user	(1) (4)
cst – Constituent Signal Type	RTRV	PROTNSW	user	(1) (4)
actunit – Active Unit State	RTRV	PROTNSW	state	(3) (4)
swreq – Switch Request State	RTRV	PROTNSW	state	(3) (4)
actunitaid – Active Unit State AID	RTRV		state	(3) (4)
protnsiglvl – Protection Fault Status	RTRV		state	(4)
wkgsiglvl – Worker Fault Status	RTRV		state	(4)

Note 1: Initial value has to be specified as part of **ENT-CRS**/**ENT-ROLL** command.

Note 2: Parameter is called *newppgname* in the context of **ED-PROTN-GRP** command.

Note 3: Updated either via protection switch commands (**OPR-PROTNSW**, **RLS-PROTNSW**) or autonomous switch events.

Note 4: Parameter may change dynamically if path-protection uses tributaries of ports in the adaptive mode.

Note 5: Parameter not supported for low order path protection groups.

List of commands

Autonomous notifications

1+1 equipment protection group

If the second switch circuit pack is auto created, the 1+1 switch equipment protection group entity shall be created. This entity creation shall be reported by means of **ENT-PROTN-GRP** notification.

Auto creation of the second switch circuit pack

If the second switch circuit pack is auto created, the 1+1 switch equipment protection group entity is created. This shall be reported by means of **ENT-PROTN-GRP** notification for 1+1 equipment protection group. The initial protection state shall be reported by means of a protection switch notification for the 1+1 equipment protection group.

Protection switch

A protection switch notification (that is, protection state change) is triggered by means equipment fail condition change as specified for the corresponding protection state machine.

Deletion of the second switch circuit pack

If the second switch circuit pack is deleted, the 1+1 switch equipment protection group entity shall be deleted. This entity deletion shall be reported by means of DLT-PROTN-GRP notification.

Deletion of the second controller circuit pack

If the second switch circuit pack is deleted, the 1+1 switch equipment protection group entity shall be deleted. This entity deletion is reported by means of DLT-PROTN-GRP notification for the 1+1 protection group.

Multiplex section/Line protection group

Protection switch and related events

A protection switch notification (i.e. protection state change) is triggered by means of

- fault condition change
- wait to restore timer expiry
- switch command received via APS message as specified for the corresponding protection state machine.

A protection switch notification shall be triggered upon a change of the Secondary Section Identifier (spec block parameter ’ssi’). This applies to MSP 1+1 optimized only. This requirement gives additionally to other requirements a further trigger for the autonomous notification for a MSP protection group. It does not require necessarily a separate notification beside other notifications of the same MSP protection group at the same time.

2-/4-fibre MS-SPRing/BLSR protection group

Protection switch

A protection switch notification (that is, protection state change) is triggered by means of

- fault condition change
- wait to restore timer expiry
- switch command received via APS message

as specified for the corresponding protection state machine.

Path protection group

Protection switch

A protection switch notification (that is, protection state change) is triggered by means of

- fault condition change
- wait to restore timer expiry

as specified for the corresponding protection state machine.

Note that a path protection switch notification may be suppressed if the WTR timer expiry notification follows immediately.

Protection log

Overview

Purpose

Each NE stores protection switch events in a protection log. The NE administrates the log autonomously. Therefore only a retrieve operation is defined for log the entity.

Contents

List of commands

Autonomous notifications

System start-up with default DB

The system will start up with an empty protection log as part of the default DB.

System start-up after DB transformation

The system will start up with an empty protection log after a database transformation has taken place in context of an SW upgrade.

Ring map

Overview

Purpose

Each NE within a 2-/4fibre MS-SPRing/BLSR ring will maintain a ring map entity for each MS-SPRing/BLSR protection group entity. The ring map either will be discovered automatically by the NEs (default) or manually be entered by the user (see also parameter ard in "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)).

The ring map consists of:

- the nodes within the ring
- the links connecting the nodes.

If the automatic ring discovery is disabled and the ring map is entered manually, the links connecting the nodes may not be retrievable. The ring map can be edited manually with ED-MAP-RING and is then stored in volatile memory. With SET-MAP-RING, the new ring map is written to NVM and activated.

Contents

List of commands

Circuit map

List of commands

9 TL1 messages - performance monitoring

Overview

Purpose

This section contains the TL1 interface definition for the domain “performance monitoring”.

Contents

PM register/bin

Overview

Purpose

This section contains the TL1 interface definitions listed below.

Contents

PM parameters

PM parameter overview

This section gives an overview of all PM parameters and the associated TCA profiles. Overview tables are given for the entities supporting of PM parameters:

- SONET ports/tributaries
- SDH ports/tributaries
- Ethernet ports
- DS3 ports (including the virtual TransMUX ports)
- OTN ports.
- DS1 tributaries

Note: In Ethernet, DS3, and Transparency application, the tributaries are acting as path terminations.

Each table consists of the following rows:

- PM parameter short name (as defined in the *montype* parameter)
- PM parameter long name
- associated TCA profile type (as defined in the *pftype* parameter).

Note: If thresholding is not applicable for a PM parameter, the value 'None' is specified as associated profile type.

SONET PM parameter table

Short name	Long name	Profile type
CVS	Code Violation Section	SECTION-LINE
ESS	Errored Second Section	SECTION-LINE
SESS	Severely Errored Second Section	SECTION-LINE
SEFS-S	Severely Errored Framing Second Section	SECTION-LINE
LOSS-S	LOS Second Section	None
CVL	Code Violation Line	SECTION-LINE
ESL	Errored Second Line	SECTION-LINE
SESL	Severely Errored Second Line	SECTION-LINE
UASL	Unavailable Second Line	SECTION-LINE
FECC-L	Forward Error Correction Corrected Line	SECTION-LINE
AISS-L	AIS Second Line	None
FC-L	Failure Count Line	None
CV-LFE	Code Violation Line Far End	SECTION-LINE

Short name	Long name	Profile type
ES-LFE	Errored Second Line Far End	SECTION-LINE
SES-LFE	Severely Errored Second Line Far End	SECTION-LINE
UAS-LFE	Unavailable Second Line Far End	SECTION-LINE
FC-LFE	Failure Count Line Far End	None
CVP	Code Violation Path	PATH
ESP	Errored Second Path	PATH
SESP	Severely Errored Second Path	PATH
UASP	Unavailable Second Path	PATH
FC-P	Failure Count Path	None
CV-PFE	Code Violation Path Far End	PATH
ES-PFE	Errored Second Path Far End	PATH
SES-PFE	Severely Errored Path Line Far End	PATH
UAS-PFE	Unavailable Second Path Far End	PATH
FC-PFE	Failure Count Path Far End	None
PPJC-PGEN	Positive Pointer Justification Count Path Generated	PATH
NPJC-PGEN	Negative Pointer Justification Count Path Generated	PATH
PPJC-PDET	Positive Pointer Justification Count Path Detected	PATH
NPJC-PDET	Negative Pointer Justification Count Path Detected	PATH

SDH PM parameter table

Short Name	Long Name	Profile Type
RS-N-ES	RS Near End Errored Second	RS-MS
RS-N-SES	RS Near End Severely Errored Second	RS-MS
RS-N-BBE	RS Near End Background Block Error	RS-MS
RS-N-UAS	RS Near Unavailable Second	RS-MS
MS-N-ES	MS Near End Errored Second	RS-MS
MS-N-SES	MS Near End Severely Errored Second	RS-MS
MS-N-BBE	MS Near End Background Block Error	RS-MS
MS-N-UAS	MS Near Unavailable Second	RS-MS
MS-N-FECC	MS Near End Forward Error Correction Corrected	RS-MS
MS-F-ES	MS Far End Errored Second	RS-MS
MS-F-SES	MS Far End Severely Errored Second	RS-MS
MS-F-BBE	MS Far End Background Block Error	RS-MS

Short Name	Long Name	Profile Type
MS-F-UAS	MS Far Unavailable Second	RS-MS
ODU-N-ES	ODU Near End Errored Second	ODU
ODU-N-SES	ODU Near End Severely Errored Second	ODU
ODU-N-BBE	ODU Near End Background Block Error	ODU
ODU-N-UAS	ODU Near End Unavailable Second	ODU
ODU-F-ES	ODU Far End Errored Second	ODU
ODU-F-SES	ODU Far End SeverelyErrored Second	ODU
ODU-F-BBE	ODU Far End Background Block Error	ODU
ODU-F-UAS	ODU Far End Unavailable Second	ODU
VC-N-ES	VC Near Unavailable Second	HOVC
VC-N-SES	VC Near End Errored Second	HOVC
VC-N-BBE	VC Near End Severely Errored Second	HOVC
VC-N-UAS	VC Near End Background Block Error	HOVC
VC-F-ES	VC Far End Errored Second	HOVC
VC-F-SES	VC Far End Severely Errored Second	HOVC
VC-F-BBE	VC Far End Background Block Error	HOVC
VC-F-UAS	VC Far Unavailable Second	HOVC
PJCPOS	Pointer Justification Count Positive	HOVC
PJCNEG	Pointer Justification Count Negative	HOVC

Ethernet PM parameter table for GE1

Short Name	Long Name	Profile Type
CBR	Count (number) of Megabytes (10^6 octets) Received	None
CBS	Count (number) of Megabytes (10^6 octets) Sent	None
PDE	Number of Errored Packets Dropped	ENET

Alternative Ethernet PM parameter table

Short Name	Long Name	Profile Type
EINB	Number of bytes received	None
EONB	Number of bytes sent	None
EINB	Number of bytes received	None
EONF	Number of packets sent	None

Short Name	Long Name	Profile Type
EDFE	Packets dropped due to error	ENET
EDFC	Packets dropped due to congestion	ENET

DS3 PM parameter table

Short Name	Long Name	Profile Type
CVL	Code Violation Line	DS3
ESL	Errored Second Line	DS3
ESA-L	Errored Second Type A, Line	DS3
ESB-L	Errored Second Type B, Line	DS3
SESL	Severely Errored Second Line	DS3
LOSS-L	LOS Second Line	None
CVP	Code Violation Path	DS3
ESP	Errored Second Path	DS3
ESA-P	Errored Second Type A, Paht	DS3
ESB-P	Errored Second Type B, Paht	DS3
SESP	Severely Errored Second Path	DS3
UASP	Unavailable Second Path	DS3
FC-P	Failure Count Path	None
SEFS-P	Severely Errored Framinf Second Path	None
SAS-P	SEF/AIS Seconds Path	DS3
AISS-P	AIS Second Path	None
CV-PFE	Code Violation Path Far End	DS3
ES-PFE	Errored Second Path Far End	DS3
ESA-PFE	Errored Second Type A, Path Far End	DS3
ESB-PFE	Errored Second Type A, Path Far End	DS3
SES-PFE	Severely Errored Second Path Far End	None
UAS-PFE	Unavailable Second Path Far End	DS3
FC-PFE	Failure Count Path Far End	None
SAS-PFE	SEF/AIS Seconds Path Far End	DS3

OTN PM parameter table

Short Name	Long Name	Profile Type
OTN-LOSS	OTN LOS Seconds	None
OTN-FECC	OTN MS OTN FEC Corrected Errors	None

DS1 PM parameter table

Short Name	Long Name	Profile Type
CVP	Code Violation Path	DS1
ESP	Errored Second Path	DS1
SESP	Severely Errored Second Path	DS1
UASP	Unavailable Second Path	DS1
SAS-P	SEF/AIS Seconds Path	DS1
AISS-P	AIS Second Path	None

Details for PM parameters with no thresholding

Below those PM parameters are described which do not support thresholding. All other parameters which support thresholding are described in the Output Parameters section of RTRV-TCA-PROF.

SONET PM parameters

1. qhaissl
   Description: 15-minute AISS-L bin for SONET Line.
   The 15-minute AISS-L bin shall be supported for OC-n Line. The 15-minute AISS-L bin shall be supported for EC-1 Line.
   Type: integer
   Range: 0-900
2. dayaissl
   Description: Day AISS-L bin for SONET Line.
   The Day AISS-L bin shall be supported for OC-n Line. The Day AISS-L bin shall be supported for EC-1 Line.
   Type: integer
   Range: 0-86400
3. qhlosss
   Description: 15-minute LOSS-S bin for ONET Section.
   The 15-minute LOSS-S bin shell be supported for OC-n Section. The 15-minute LOSS-S bin shell be supported for EC-1 Section.
   Type: integer
   Range: 0-900

4. daylosss
   Description: Day LOSS-S bin for SONET Section.
   The Day LOSS-S bin shall be supported for OC-n Section. The Day LOSS-S bin shall be supported for EC-1 Section.
   Type: integer
   Range: 0-86400
5. qhfcl
   Description: 15-minute FC-L and FC-LFE bin for OC-n lines.
   The 15-minute FC-L and FC-LFE bin shall be supported for OC-n Line. The 15-minute FC-L and FC-LFE bin shall be supported for EC-1 Line.
   Type: integer
   Range: 0-900
6. dayfcl
   Description: Day FC-L and FC-LFE bin for SONET Line.
   The Day FC-L and FC-LFE bin shall be supported for OC-n lines. The Day FC-L and FC-LFE bin shall be supported for EC-1 lines.
   Type: integer
   Range: 0-86400
7. qhfcp
   Description: 15-minute FC-P and FC-PFE bin for STS1 and STS1-xC (x=3, 12, 48, 192) tributaries
   Type: integer
   Range: 0-900
8. dayfcp
   Description: Day FC-P and FC-PFE bin for STS1 and STS1-xC (x=3, 12, 48, 192) tributaries
   Type: integer
   Range: 0-86400

Ethernet PM parameters

1. qhcbr
   Description: 15-minute CBR for a Ethernet port
   Type: integer
   Range: 0-110000
2. daycbr
   Description: Day CBR for a Ethernet port.
   Type: integer
   Range: 0-10560000
3. qhcbs
   Description: 15-minute CBS for a Ethernet port
   Type: integer
   Range: 0-110000

4. daycbs
   Description: Day CBS for a Ethernet port.
   Type: integer
   Range: 0-10560000
5. qheinb1ge
   Description: 15 minute EINB for a 1 GbE LAN and WAN port
   Type: integer
   Range: 0-112500000000
6. dayeinb1ge
   Description: Day EINB for a 1 GbE LAN and WAN port
   Type: integer
   Range: 0-10800000000000
7. qheonb1ge
   Description: 15 minute EONB for a 1 GbE LAN and WAN port.
   Type: integer
   Range: 0-112500000000
8. dayeonb1ge
   Description: Day EONB for a 1 GbE LAN and WAN port.
   Type: integer
   Range: 0-10800000000000
9. qheinf1ge
   Description: 15 minute EINF for a 1 GbE LAN and WAN port
   Type: integer
   Range: 0-1371951220
10. dayeinf1ge
    Description: Day EINF for a 1 GbE LAN and WAN port
    Type: integer
    Range: 0-131707317073
11. qheonf1ge
    Description: 15 minute EONF for a 1 GbE LAN and WAN port
    Type: integer
    Range: 0-1371951220
12. dayeonf1ge
    Description: Day EONF for a 1 GbE LAN and WAN port
    Type: integer
    Range: 0-131707317073
13. qheinb10ge
    Description: 15 minute EINB for a 10 GbE LAN and WAN port
    Type: integer
    Range: 0-1125000000000

14. dayeinb10ge
Description: Day EINB for a 10 GbE LAN and WAN port
Type: integer
Range: 0-108000000000000
15. qheonb10ge
Description: 15 minute EONB for a 10 GbE LAN and WAN port
Type: integer
Range: 0-1125000000000
16. dayeonb10ge
Description: Day EONB for a 10 GbE LAN and WAN port
Type: integer
Range: 0-108000000000000
17. qheinf10ge
Description: 15 minute EINF for a 10 GbE LAN and WAN port
Type: integer
Range: 0-13719512195
18. dayeinf10ge
Description: Day EINF for a 10 GbE LAN and WAN port
Type: integer
Range: 0-1317073170732 Default
19. qheonf10ge
Description: 15 minute EONF for a 10 GbE LAN and WAN port
Type: integer
Range: 0-13719512195
20. dayeonf10ge
Description: Day EONF for a 10 GbE LAN and WAN port
Type: integer
Range: 0-1317073170732

DS3 PM Parameters

1. qhaissp
Description: 15-minute AISS-P bin for DS3
Type: integer
Range: 0-900
2. dayaissp
Description: Day AISS-P bin for DS3.
Type: integer
Range: 0-86400
3. qhfcp
Description: 15-minute FC-P and FC-PFE bin for DS3
Type: integer
Range: 0-900

4. dayfcp
   Description: Day FC-P and FC-PFE bin for DS3
   Type: integer
   Range: 0-86400
5. qhsefsp
   Description: 15-minute SEFS-P bin for DS3.
   Type: integer
   Range: 0-900
6. daysefsp
   Description: Day SEFS-P bin for DS3
   Type: integer
   Range: 0-86400
7. qhlossl
   Description: 15-minute LOSS-L bin for DS3.
   Type: integer
   Range: 0-900
8. daylossl
   Description: Day LOSS-L bin for DS3.
   Type: integer
   Range: 0-86400

DS1 PM Parameters

1. qhaissp
   Description: 15-minute AISS-P bin for DS1.
   Type: Integer.
   Range: 0 – 900.
   Default: None.
2. dayaissp
   Description: Day AISS-P bin for DS1.
   Type: Integer.
   Range: 0 – 86400.
   Default: None.

OTN PM Parameters

1. qhotnloss
   Description: 15-minute LOSS bin for OTN
   Type: integer
   Range: 0-900
2. dayotnloss
   Description: Day LOSS bin for OTN
   Type: integer
   Range: 0-86400

3. qhotnfecc
   Description: 15-minute FECC bin for OTN
   Type: integer
   Range: 0-4294967295
4. dayotnfecc
   Description: Day FECC bin for OTN
   Type: integer
   Range: 0-4294967295

List of commands

TCA Profile

Overview

Purpose

This section contains the TL1 interface definitions listed below.

Contents

TCA Profile

General

The threshold settings for the different PM parameters supporting thresholding is provisionable by means of TCA profiles. The profiles of a certain type contains all pm parameters related to a certain entity supporting enabling/disabling of performance monitoring. Therefore the following profiles types are defined:

- SECTION-LINE - used/referenced by SONET port entities
- RS-MS - used/referenced by SDH port entities
- PATH - used/referenced by SONET and TransMUX STS1 tributary (TMASTS1) entities
- HO-VC - used/referenced by SDH tributary entities
- ENET - used/referenced by Ethernet port entities
- ODU - used/referenced by ODU/VCG entities
- DS3 - used/referenced by DS3 and TransMUX (TMAT3) port entities
- DS1 - used/referenced by DS1 tributary entities
- LO-PATH - used/referenced by SONET lower order tributary entities related to the LOXC
- LO-VC - used/referenced by SDH lower order tributary entities related to the LOXC

All threshold data of one certain profile type is sometimes also called a profile image. This term may be used in other TCA profiles commands as parameter name. It is then used as an alias for all PM thresholds parameters of the corresponding profile type.

TCA suppression can be provisioned by means of setting the corresponding profile parameter to zero.

The system supports two default sets of profiles for each type of TCA profile. One set of default profiles shall contain zero threshold value for each parameter in the profile. Another set of default profiles contains real non-zero threshold value for each parameter in the profile. The name of the zero threshold value default profile shall be “Default0” and the name of the non-zero threshold value default profile shall be “Default”.

The number of TCA profiles are limited to a specific number per profile type. This number also includes the system default profiles.

The maximum numbers are for:

- PATH SDH and PATH SONET together (higher order) 80
- RSMS SDH and section/line SONET together 30
- for ENET and ODU together 10

TCA Profile

- DS3 10
- LOXC PATH SDH and SONET together 20.
- TMUX DS1 path 10

The user is able to change the threshold values of the non-zero default profile, but will not be allowed to change its name. The user is not allowed to change the name or the threshold values of the zero default profile.

All the profile pointers are pointed to zero threshold value default profiles when the ports are provisioned. That is, thresholding is not performed as a default. All the profile pointers are pointed to non-zero threshold value default profiles when thresholding is to be performed.

Threshold parameters names are defined using the following pattern:

<bintype>[rtr]<montype>[<port/trib rate|port type|monitoring point> with:

- bintype = day (for 24h) or qh (for 15 Min)
- [rtr] = rtr is inserted if the clear threshold is mentioned
- montype = montype as defined in **RTRV-PM** or **RTRV-TMUXPM** command
- portrate = ec1, oc3, oc12, oc48, oc192, oc768, stm1, stm4, stm16, stm64, stm256, odu1, ds3
- trib rate = sts1, sts3c, sts12c, sts48c, sts192c or vc3, vc4, vc44c, vc416c, vc464c, hosts1, vt1, hovc4, lovc3, vc12, ds1
- porttype = ge, 1ge, 10ge, vcg
- monitoring point = l, p.

List of commands

System Start-up with Default DB

General

The system shall create during start-up with default DB two TCA profiles per supported profile type with profile name ″DEFAULT″ and ″DEFAULT0″. The values for profile ″DEFAULT″ shall be as specified in corresponding default requirements for each profile parameter. The values for profile ″DEFAULT0″ shall be ’0’ for each profile parameter.

10 TL1 messages - synchronization

Overview

Purpose

This section contains the TL1 commands related to the synchronization management. The synchronization commands covers types of assigning, modifying or removing and retrieving system timing attributes. Furthermore they are initiating switch requests and releasing requests.

It contains following commands and levels.

- RTRV-SYNCN
  - System Timing
  - Output Timing
  - Assigned Timing Reference
  - External Timing Input
  - External Timing Output
- SET-SYNCN
  - System Timing
  - Assigned Timing Reference
  - External Timing Input
  - External Timing Output
- OPR-SYNCNSW
- RLS-SYNCNSW.

For the system, the choice between SONET and SDH is determined by the system synchronization characteristic. The system synchronization characteristic may only be adjustable in the MCOND state. If the system synchronization characteristic is SONET, all SONET related parameter and settings are valid. All SSM messages, based on the SDH market are handled as being invalid. If the system synchronization characteristic is SDH, all SDH related parameter and settings are valid. All SSM messages, based on the SONET market are handled as being invalid.

The values of a number of parameters will be distinguished from the system synchronization characteristic. The parameter *synccrtc* in RTRV-EQPT and ED-EQPT determines whether the system synchronization characteristic, and therewith the NE, is in SDH or in SONET mode.

Contents

List of commands

Synchronization protection switch notification

Output format

Syntax: The **REPT EVT** notification has following specific syntax:

REPT EVT *modifier aid:condtype,condeff,ocrdat,ocrtm:conddescr*

Output parameters

System timing

1. Clock Mode Protection Type
   a. modifier – message modifier
      This parameter indicates on which entity the report event belongs.
      Type: Alphanumeric value set.
      Range: SYST (means System Timing).
   b. aid – access identifier
      This parameter indicates on which AID the report event belongs.
      Type: Alphanumeric value set.
      Range: SYSTEM
   c. condtype – condition type
      This parameter specifies the type of event being reported.
      Type: alphanumeric value set.
      Range: See Table 10-1, "Range of condtype (clock mode protection type)" (p. 10-5).
      Default: none
   d. conddescr – description of the condition
      This parameter is a textual description of the condition.
      Type: inner string.
      Range: See Table 10-2, "Range of conddescr (clock mode protection type)" (p. 10-5).
2. Timing Reference Protection Type
   a. modifier – message modifier
      See definition above.
   b. aid – access identifier
      This parameter indicates on which AID the report event belongs. The values all or *someprefix* all are not allowed in this context.
      Type: Alphanumeric value set.
      Range:
      – External_timing_ref_AID
      – Line_timing_ref_AID
      – NONE.
      The value NONE shall be reported when the last timing reference is failed.

c. condtype – condition type

This parameter specifies the type of event being reported.

Type: alphanumeric value set.

Range: See Table 10-3, "Range of condtype (timing reference protection type)" (p. 10-5).

d. conddescr – description of the condition

This parameter is a textual description of the condition.

Type: inner string.

Range: See Table 10-4, "Range of conddescr (timing reference protection type)" (p. 10-6).

Table 10-1 Range of condtype (clock mode protection type)

Selection State	Attribute	Condtype
free running	clkmod = FREE-RUNNING clkmod_stat = NORMAL	FRNGSYNC
forced holdover	clkmod = LOCKED clkmod_stat = FORCED-HOLDOVER	HLDOVRSYNC
autonomous holdover	clkmod = LOCKED clkmod_stat = AUTONOMOUS-HOLDOVER	HLDOVRSYNC
locked	clkmod = LOCKED clkmod_stat = NORMAL	SYNCCLK

Table 10-2 Range of conddescr (clock mode protection type)

Selection State	Conddescr
free running	Protection switch, clock mode, Freerunning
forced holdover	Protection switch, clock mode, Forced Holdover
autonomous holdover	Protection switch, clock mode, Holdover
locked	Protection switch, clock mode, Locked

Table 10-3 Range of condtype (timing reference protection type)

Selection State / Lockout State	Attribute	Condtype
manual	refsw_stat = MANUAL	PS
forced	refsw_stat = FORCED	PS
no request	refsw_stat = NO-REQUEST	PS
lockout	ref_lock = LOCKOUT	INHSWPR
no lockout	ref_lock = NO-LOCKOUT	INHSWPR

Table 10-4 Range of conddescr (timing reference protection type)

Selection State / Lockout State	Conddescr
manual	Protection switch,sync reference, manual switch
forced	Protection switch,sync reference, forced switch
no request	Protection switch,sync reference, no Request
lockout	Protection switch,sync reference, lockout
no lockout	Protection switch,sync reference, no lockout

Output timing (only for SDH)

1. Output Timing Reference Protection Type
   a. modifier – message modifier
      This parameter indicates on which entity the report event belongs.
      Type: Alphanumeric value set.
      Range: OUTT (means Output Timing).
   b. aid – access identifier
      This parameter indicates on which AID the report event belongs. The values all or *someprefix* all are not allowed in this context.
      Type: Alphanumeric value set.
      Range:
      – Line_timing_ref_AID
      – NONE.
      The value NONE shall be reported when the last timing reference is failed.
   c. condtype – condition type
      This parameter specifies the type of event being reported.
      Type: alphanumeric value set.
      Range: See Table 10-5, "Range of condtype (Output timing reference protection type)" (p. 10-6).
   d. conddescr – description of the condition
      This parameter is a textual description of the condition.
      Type: inner string
      Range: See Table 10-6, "Range of conddescr (Output timing reference protection type)" (p. 10-7).

Table 10-5 Range of condtype (Output timing reference protection type)

Selection State / Lockout State	Attribute	Condtype
manual	orefswr_stat = MANUAL	PS
forced	orefswr_stat = FORCED	PS

Table 10-5 Range of condtype (Output timing reference protection type) (continued)

Selection State / Lockout State	Attribute	Condtype
no request	orefswr_stat = NO-REQUEST	PS
lockout	sclkref_lockstat = LOCKOUT	INHSWPR
no lockout	sclkref_lockstat = NO-LOCKOUT	INHSWPR

Table 10-6 Range of conddescr (Output timing reference protection type)

Selection State / Lockout State	Conddescr
manual	Protection switch, output reference, Manual switch
forced	Protection switch, output reference, Forced switch
no request	Protection switch, output reference, No Request
lockout	Protection switch, output reference, Lockout
no lockout	Protection switch, output reference, No Lockout

Example output

The following is an example of a protection switch notification for clock mode protection type:

```
   LT-UNITE 00-11-09 12:00:00
A  000 REPT EVT SYST
   ″SYSTEM:HLDOVRSYNC,TC,11-09,12-00-00,,,:\″Protection switch,clock
      mode,Forced Holdover\″″
;
```

The following is an example of a protection switch notification for timing reference protection type:

```
   LT-UNITE 00-11-09 12:00:00
A  000 REPT EVT SYST
   ″EXTREF1:PS,TC,11-09,12-00-00,,,:\″Protection switch,sync
      reference,Manual switch\″″
;
```

The following is an example of a protection switch notification for timing reference protection type when the last reference has failed:

```
   LT-UNITE 07-01-26 14:41:00
A  000 REPT EVT SYST
   ″NONE:PS,TC,07-01-26,14-40-30,,,:\″Protection switch,sync
      reference,No Request\″″
```

```
;
```

The following is an example of a protection switch notification for output timing reference protection type:

```
   LT-UNITE 00-11-09 12:00:00
A  000 REPT EVT OUTT
   ″LINE2:INHSWPR,SC,11-09,12-00-00,,,:\″Protection switch,output
      reference,Lockout\″″
;
```

The following is an example of a protection switch notification for output timing reference protection type when the last timing reference has failed:

```
   LT-UNITE 07-01-26 14:42:00
A  000 REPT EVT OUTT
   ″NONE:PS,TC,07-01-26,14-41-30,,,:\″Protection switch,output
      reference,No Request\″″
;
```

Autonomous notifications

System initialization with default values

After first system initialization for all parameters the described default values are valid and no timing reference is assigned. No database change notification will be returned during database auto creation.

System initialization after system synchronization characteristic change

It is possible to change the system synchronization characteristic later on. For this case, the system will behave as a new created node. All assigned timing references are un-assigned before and after changing in the MCOND state the system will start-up with the default values. No database change notification will be returned after changing the system synchronization characteristic and reset to default values.

Autonomous clock switch

If the provisioned clockmod = LOCKED and a failure of all timing references active, the system switches in autonomous-holdover state and return a report event and a report alarm notification to the management system. If at least one fault-free timing reference is available, the system switches back to the locked state autonomously. The following parameter are affected by the autonomous clock mode switch: clkmod_stat = [AUTONOMOUS-HOLDOVER, NORMAL].

Autonomous reference switch for system timing

The autonomous reference switching is initiated by the failure of a reference. The autonomous reference switching results also after a SSM change of the active reference or of any of the standby references. The choice for the next active timing reference primarily depends on provisioned QL values if QL mode is enabled and subsequent on provisioned priorities for each reference. In consequence of identical QL values and priorities the system supports non-revertive switching. The autonomous reference switch returns a protection switch notification for System Timing. The following parameter are affected by the autonomous timing reference switch: active_ref = [External_timing_ref_AID, Line_timing_ref_AID, NONE].

Autonomous reference switch for output timing (only for SDH)

The autonomous reference switching is initiated by the failure of a reference. The autonomous reference switching might also switch after a SSM change. The choice for the next active timing reference is primarily depending on provisioned QL values if QL mode is enabled and subsequent on provisioned priorities for each reference. In consequence of similar QL values and priorities the system supports non-revertive switching. The autonomous reference switch returns a protection switch notification for Output Timing. The following parameter are affected by the autonomous timing reference switch: active_oref_stat = Line_timing_ref_AID, NONE.

Synchronization instances

The following instances are created during system initialization:

- one system timing entity
- one output timing entity (only for SDH)
- two timing reference entities for external timing input
- six timing reference entities for line timing for

SDH only:

- one external timing output port entity

for SONET only:

- two external timing output port entities.

11 TL1 messages - software & database management

Overview

Purpose

This section contains the TL1 interface definition for the domain *software and database management*.

Background information:

In the Non Volatile Memory (NVM) of the NE, there are two regions: current and previous. Each region contains a software generic (code and static data) and a database.

The properties of the databases and the software generics can be retrieved by RTRV-PRMTR-DATA and RTRV-PRMTR-SFTWR.

The restore option of CPY-MEM is used to copy a database from a remote file store into previous NVM and then to install it as the working database. The backup option of CPY-MEM is used to copy the database that is in current NVM to a remote file store.

The download option of CPY-MEM is used to download new generics to previous NVM. Installation via APPLY causes the contents of previous NVM to be installed. Following installation, a ″pointer″ is changed so that what is identified as current NVM contains the currently executing generic and previous NVM contains the previous generic.

Contents

List of commands

12 TL1 messages - data communications network

Overview

Purpose

This section lists the TL1 interface definition for the domain *data communications network*.

Contents

General DCN parameters

Overview

Purpose

This section describes TL1 messages for DCN which have effects on system wide DCN configurations.

Contents

List of commands

DCC terminations

Overview

Purpose

This section lists TL1 commands for section RS-DCC and Line MS-DCC: provisioning, modification and status retrieval.

Contents

List of commands

- "RTRV-FECOM: Retrieve DCC related parameters" (p. 3-2407)
- "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
- "ED-FECOM notification" (p. 3-402)
- "ENT-FECOM notification" (p. 3-1578)
- "DLT-FECOM notification" (p. 3-197)

Transparent DCC cross-connection

Overview

Purpose

Transparent DCC cross-connections allows to cross-connect DCC channels (either DCC-R or DCC-M) of an OPn/EPn on the XCn without adding/dropping them to/from the OHI bus to/from the CTL/DCF. As such there is no processing on the DCF for those channels a transparent DCC cross-connection allows to transparently pass-through certain DCC channel without any processing by the system. The transparent DCC cross-connections are complete independent from the transmission cross C-connection. The transparent DCC cross-connections are always bidirectional. The system supports up to 25 bidirectional transparent DCC cross-connections.

Contents

List of commands

LAN

Overview

Purpose

This section lists TL1 commands applicable to the LAN for *WaveStar*® CIT respectively *Navis*® Optical EMS access.

Contents

List of commands

Autonomous notification

Start-up with default database

The event ″Start-up with default DB″ causes that the three LAN instances with AIDs lan1, lan2, lan3 are created. No notifications shall be sent.

OSI network layer

Overview

Purpose

This section lists TL1 commands applicable to layer 3 of the OSI stack.

Contents

List of commands

Autonomous notification

Start-up with default database

The event ″Start-up with default DB″ causes that the OSI node 1 with AID osinode1 is created.

OSI transport layer and TARP

Overview

Purpose

This section lists TL1 commands applicable to layer 4 of the OSI stacks.

Contents

List of commands

TCP/IP parameters

Overview

Purpose

This section lists TL1 commands applicable to the TCP/IP stack.

Contents

List of commands

IP Map

Overview

Purpose

This section describes commands dealing with the IP map:

- the IP map provides T-TD access to the NE via one of the LANs that is assigned to node 1. Only managers registered in the IP map are allowed to access the NE via TCP/IP unless the wildcard (255.255.255.255) is entered into the list. In this case the access restriction is disabled.
- historically, the application context identifier for TL1 over OSI connections has been used to tailor the characteristics of the TL1 session (e.g. which events are reported using TL1 notifications) to the calling manager.
- ADM40G bases the characteristics of a TL1 session on the user name passed in the ACT-USER command rather than the application context identifier.
- still, for backwards compatibility, the feature of having the application context identifier being determined by the address of the calling manager is required on the T-TD.

Contents

List of commands

MAPs

Overview

Purpose

This section lists TL1 commands applicable to Maps.

Contents

List of commands

Control Channel Provisioning

Overview

Purpose

This section lists TL1 commands applicable to control channel provisioning.

Contents

List of commands

13 Optical network navigation system (ONNS)

Overview

Purpose

This chapter contains detailed information about the TL1 commands related to the so-called Optical Network Navigation System (ONNS) feature.

The ONNS system component name is often shortened to SNN or NN.

Contents

ONNS system parameters

Overview

Purpose

This section lists information about the TL1 commands related to the ONNS system parameters.

Contents

List of commands

ONNS data repositories

Overview

Purpose

This section lists information about the TL1 commands related to the ONNS data repositories.

Contents

List of commands

Path

Overview

Purpose

This section contains information about the TL1 commands related to the Path.

Contents

List of commands

Connection and path state definitions

Overview

The Connection connStatus and the pathState variables are related, and the connStatus can be derived from the pathState. The two tables below show the possible states.

The column State Type characterizes the codes further:

- ES: connection/path is in a stable and active End State
- FR: Final Report state (indicates that the connection/path is no longer active)
- TS: Transient State
- RQ connection/path on Restoration Queue.

connStatus Codes

TL1 *connStatus* code	Description	State Type
CON&OK	Normal Connection Status	ES
CON&FAILED	Connection failure	ES
CON&NO-IN	No Input, source or destination (Initial setup)	ES
CON&FAILED&RSTRFL	Restoration attempt failed / connection going to the Restoration Queue	TS
CON&TRDOWN	Connection Delete (In Progress)	TS
DLT	Connection Deleted	FR
CON&DLT-FL	Conn Delete Failed, path not deleted	ES
CON&OK&RSTRD	Auto-Reroute Revertive Connection Status (after initial restoration)	ES
NOCON&SETUP	Initial connection setup in progress	TS
NOCOM&PREMPTD	Connection has been disconnected from original path	TS
CON&OK&WTORV	AUTORV Path ready, waiting for WTR (wait-to-revert) timer to revert to original path	TS
NOCON&WTORSTR	Preempted path removed & Connection has been added to the Restoration Queue.	ES / RQ
CON&FAILED&WTORSTR	Auto Connection failed & has been added to the Restoration Queue.	ES / RQ
CON&SETUP	Initial connection setup completed, waiting for check of signal	TS
NOCON&SETUPFL	Initial connection setup failed	FR
NOCON	Connection has been disconnected from original path by maintenance operation	TS

TL1 *connStatus* code	Description	State Type
CON&STBY	Protecting connection not being allocated for protecting a working one	ES
CON&USED	Protecting connection not being allocated for protecting a working one	ES
CON&UNKNOWN	Connection Status is not known after a recovery until the fault status is retrieved from the z-node	TS
CON&OK&DEGRADED	Path failure of a permanent 1+1 TS connection can't be restored by a new path	TS

PathState Codes

TL1 *pathState* code	Description	State Type
CON&OK	Normal Path Status	ES
CON&FAILED	Path failure	ES
CON&TRDOWN	Path Delete in progress	TS
DLT	Path Delete complete	FR
CON&NO-IN	No Input, source or destination, during initial setup	ES/TS
CON&OK&RSTRD	Auto-Reroute Revertive Restoration	ES
NOCON&SETUP	Initial connection setup in progress	TS
CON&DLT-FL	Path delete failed	ES
NOCON&PREMPTD	Path has been disconnected from original EDGE port and will be re-used on another restored connection	TS
CON&OK&WTORV	AUTORV Path ready, waiting for WTR (wait-to-revert) timer to revert to original path	TS
NOCON	Pre-calculated restore path available for use	ES
CON&SETUP	Initial connection setup completed, waiting for signal confirmation from the egress node	TS
CON&FAILED&WTORSTR	Restoration attempt failed and will be restored when a new path is available	ES
NOCON&SETUPFL	Initial connection setup failed	ES
RSVD&OK	Normal Path Status while associatedto a protecting connection	ES
RSVD&FAILED	Path Failure while associated to a protecting connection	ES

TL1 *pathState* code	Description	State Type
RSVD&DLT-FL	Path delete failed for a path associated to a protecting connection	ES
RSVD&SETUPFAIL	Initial connection setup failed for protecting connection	ES
RSVD&TRDOWN	Path Delete in progress for path associated to protecting connection	TS
RSVD&SETUP	Initial connection setup completed, for path associated to protecting connection, waiting for signal confirmation from the egress node	TS
CON&UNKNOWN	Path State is not known after a recovery until the fault status is retrieved from the z-node.	TS
RSVD&UNKNOWN	Path State for a path associated to a protecting connection is not known after a recovery until the fault status is retrieved from the z-node.	TS

Network optimization

Overview

Purpose

This section contains information about the TL1 commands related to the network optimization.

Contents

List of commands

Maintenance

Overview

Purpose

This section contains information about the TL1 commands related to the maintenance.

Contents

List of commands

UNI commands

Overview

Purpose

This section contains information about the TL1 commands related to the UNI feature.

Contents

List of commands

ENNI commands

Overview

Purpose

This section contains information about the TL1 commands related to the ENNI feature.

Contents

List of commands

14 Error codes

List of error codes

Error code	Meaning (validation rule)	Related to command
DNC - Input, Data Not Consistent, No BLSR/MS-SPRing inter-working with low-order traffic	LO Substructure creation not allowed on BLSR/MS-SPRing ports	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
DNC - Input, Data Not Consistent, revertive mode not supported for MSP Protection Type 1+1 and a port with the indicated rate	No MSP/APS 1+1 revertive protection for bitrate STM-256/OC-768	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
DNV - Input, Data Not Valid, DCC on DS3 Ports not allowed	No DCC on DS3 ports	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510); "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
DNV - Input, Data Not Valid, DCC on EC1 Ports not allowed	No DCC on EC1 ports	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510); "ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
E001 - srvID - InSufficient Network Resources for connection setup	Insufficient network resources	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E001 - srvID - InSufficient Network Resources for connection setup	Insufficient network resources for alternate route	"ED-PATH-ROLL: Modify ONNS connection route" (p. 3-781)
E002 - SICE - Status, Inter-Processor Communication Error caused cmd failure	Access to the database is not possible I	"RTRV-NADR: Retrieve ONNS network (domain) address data repository" (p. 3-2539)
E002 - SICE Error - Status, Inter-Processor Communication Error caused cmd failure	Invalid componentID parameter II	"RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
E003 - SNTT - Status, Network Topology database Timed out	Access to the database is not possible II	"RTRV-NADR: Retrieve ONNS network (domain) address data repository" (p. 3-2539)

Error code	Meaning (validation rule)	Related to command
E004 - srvID - InSufficient Network Resources for Alternate Path	Insufficient network resources for alternate route	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E004 - srvID - InSufficient Network Resources for Alternate Path	Insufficient Network Resources for Alternate Route	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E020 - Maximum Transport Delay Exceeded (by calculated path)	Maximum delay errors on calculated path	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753); "ED-PATH-ROLL: Modify ONNS connection route" (p. 3-781)
E020 - srvID - Maximum Transport Delay Exceeded,errorData	Maximum Delay Exceeded by calculated path	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E030 - SMcon135*7.92468 - Syntax, Parameter Data Imvalid,Invalid_ parameter=SrvID	Invalid srvID	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E030 - SPDI - Syntax, Parameter Data Invalid, invalid_parameter= fieldName	ED-PATH request denied - Parameter invalid	"ED-PATH-ROLL: Modify ONNS connection route" (p. 3-781)
E030 - srvID - Syntax, Parameter Data Invalid,invalid_parameter= fieldName	ENT-PATH syntax, parameter data invalid	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E030 - Syntax, Parameter Data Invalid, invalid_parameter	ED-PATH-PROT request denied - Parameter invalid	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E040 - srvID - Syntax, Data Not Consistent	ENT-PATH syntax, data not consistent	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E050 - IICG - Input, Invalid connection ID for connection group/conn ID	Retrieve path - Invalid connection ID for group retrieval	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E081 - IIEP - Input, Invalid ingress / Egress Port,errorData	NN PortID parameter value is not valid	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E082 - Status, NNI connPoint / port Unavailable ID=<ASTN node ID> <ConnPtID>	Port related problems	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E150 - Status, Path Protection Group parameters Not Provisioned	ED-PATH-PROT Path Protection Group Parameter transfer failed	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E160 - srvID - XConn Connect Failure, Port does not exist or is TRADITIONAL	Cross-connect connect failure	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E220 - 135.13.7.9-48 - Input, ConnectionID Non-Existent	Path ConnectionID does not exist	"DLT-PATH: Delete ONNS path connection" (p. 3-222)
E220 - 13513794 - Input, ConnectionID Non-Existent	connID Non-Existant (ICNE)	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)

Error code	Meaning (validation rule)	Related to command
E222 - IIDM srvID - Input, Invalid deletion of connection of protection	Connection is member of an M:N protected connection group	"DLT-PATH: Delete ONNS path connection" (p. 3-222)
E223 - SPCU 135.7.9.2-35 - Status, Protection Connection in Use, removal of the M:N worker connection aborted	Protection connection of an M:N protection group in use	"DLT-PATH: Delete ONNS path connection" (p. 3-222)
E224 - IIDC srvID - Input, Invalid Deletion of a Connection of a switched connection group	Connection is member of a switched VCG connection group	"DLT-PATH: Delete ONNS path connection" (p. 3-222)
E225 - IIGD srvID - Input, Invalid Grouped Deletion of EXT Connection	Grouped connection deletion on VCG EDGE affects EXT Connection	"DLT-PATH: Delete ONNS path connection" (p. 3-222)
E226 - IDNA - Input, Retrieve Detail_Route or Detail_State Not allowed for RTRV-PATH ALL	Error Responses Specific to RTRV-PATH	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E235 - Input, Invalid CoMmand, Command not found or recognized	Invalid command	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E240 - Syntax, InValid Punctuation	Syntax, Invalid punctuation	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E246 - Syntax, Parameter Not Implemented	Syntax invalid or unimplemented parameter	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E250 - Input, Invalid TArget Identifier	Invalid target identifier	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E260 - Input, Invalid ACcess identifier	Invalid access identifier	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E265 - Input, Invalid Correlation Tag	Invalid correlation tag (CTAG) or RequestID	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E280 - Status, All Resources Busy	Status, All Resources Busy	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E282 - SCSI - Status, Conn State Invalid for this Operation	Retrieve path with detailed route option - No path ID associated to ONNS connection	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E282 - SCSI Error - Status, Conn State Invalid for this Operation	Switch Back behavior modification	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E283 - SOPI - Status, Operation Invalid for Protection Type MBYNPRT	Retrieve path with detailed state option - Operation Invalid for Protection Type MBYNPRT	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E290 - Status, Requested Operation Failed	Requested Operation Failed (SROF)	"RTRV-PATH: Retrieve ONNS connection data" (p. 3-2698)
E360 - SPMF - Status, Path Modify Failed due to invalid protection type change	Protection modification for External Connections	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)

Error code	Meaning (validation rule)	Related to command
E511 - IINO - Input, Invalid NOde_ ID	Invalid componentID parameter I	"RTRV-NADR: Retrieve ONNS network (domain) address data repository" (p. 3-2539)
E511 - IINO - Input, Invalid NOde_ ID	Invalid componentID parameter II	"RTRV-NTDR: Retrieve ONNS network topology data repository" (p. 3-2588)
E512 - IINI Error - Input, Invalid Neighbor node ID	Invalid componentID parameter	"RTRV-NNEVT-TCA" (p. 3-2582)
E512 - Input, Invalid Neighbor node ID	Invalid PortID or componentID parameter	"RTRV-NNDR: Retrieve ONNS neighbor nodal data repository" (p. 3-2561)
E520 - IIDD - Input, Invalid Degree of Disjointness	ENT-PATH syntax, invalid degree of disjointness value	"ENT-PATH: Create ONNS path connection" (p. 3-1701)
E520 - IIDD - Input, Invalid Degree of Disjointness for protection type N1PLUS1 or YCONN	ED-PATH-PROT syntax, invalid degree of disjointness value	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
E660 - LDOS - Local Data Out of Sync	ED-SYNCDATA-PATH not possible because of local data inconsistencies	"ED-SYNCDATA: Synchronization of ONNS data" (p. 3-1198)
E700 - ICID - Invalid Connection IDentification	OPR-MAINT-ONNS with reqType = REMLEG or REMLEG_RESV not allowed because of specified connection is not valid	"OPR-MAINT-ONNS: Maintenance actions for the ONNS application" (p. 3-2053)
E701 - DLID - Duplicate Leg IDentification	OPR-MAINT-ONNS with reqType = REMLEG or REMLEG_RESV not allowed because of both pathID and connPtID specified	"OPR-MAINT-ONNS: Maintenance actions for the ONNS application" (p. 3-2053)
E702 - ULID - Unsupported Leg IDentification	OPR-MAINT-ONNS with reqType = REMLEG or REMLEG_RESV not allowed because of invalid leg identification via pathID or connPtID	"OPR-MAINT-ONNS: Maintenance actions for the ONNS application" (p. 3-2053)
EATN - Equipage, not valid for Access Type, incorrect equipage	Incorrect Equipage (EATN)	"RTRV-HDR: Retrieve header" (p. 3-2417)
EATN - Equipage, Not Valid for Access Type, retrieved file TID mismatch	TID Mismatch in retrieved File (EATN)	"RTRV-HDR: Retrieve header" (p. 3-2417)
EATN - Status, incorrect equipage, unrecognized CP for this slot	Check whether CP is acceptable	"ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
ENEQ - Equipage, Not EQuipped	Missing Equipage (ENEQ)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ENEQ - Equipage, Not EQuipped osinode	Check for parameter osinode	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
ENEQ - Equipage, Not EQuipped, ES64U destination slot empty	No forced switch to empty slot (ES64U Protection Group);	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)

Error code	Meaning (validation rule)	Related to command
ENEQ - Equipage, Not Equipped, insufficient file space	Insufficient file space	"COPY-RFILE: Copy remote file" (p. 3-106); "CPY-MEM: Copy memory" (p. 3-114)
ENEQ - Equipage, Not EQuipped, insufficient file space	Insufficient File Space (ENEQ)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ENEQ - Equipage, Not EQuipped, LOXC destination slot empty	No forced switch to empty LOXC slot (LOXC Protection Group)	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
ENEQ - Equipage, Not EQuipped, osinode	parameter osinode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
ENEQ - Equipage, Not EQuipped, TMUX destination slot empty	No forced switch to empty slot (TransMUX Protection Group);	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
ENEQ - Equipage, Not EQuipped, XC destination slot empty	No forced switch to empty XC slot (Switching Protection Group)	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
ENRE - Equipage, Not Recognized Equipage, unrecognized CP for this slot	Check exchangeability rules for electrical packs	"ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
ENRE - Equipage, Not Recognized Equipage, unrecognized CP for this slot	Check exchangeability rules for LOXC packs	"ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
ENRE - Equipage, Not Recognized Equipage, unrecognized CP for this slot	Check exchangeability rules for optical packs	"ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
ENRE - Equipage, Not Recognized Equipage, unrecognized CP for this slot	Check exchangeability rules for XC packs	"ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
ENRE - Equipage, Not Recognized Equipage, unrecognized CP for this slot	Check whether CP is acceptable	"ED-EQPT (slot): Edit equipment (slot)" (p. 3-370)
ENRE - Equipage, Not Recognized Equipage, unrecognized module for this socket	Exchangeability III: Invalid module plugged in	"ED-EQPT (socket): Edit equipment" (p. 3-379)
ENRE - Equipage, Not Recognized Equipage, unrecognized module for this socket	Module must be part of acceptable module list	"ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
ENSG - Equipage, Not Software Generic, generic not available	Generic not available	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35); "CPY-MEM: Copy memory" (p. 3-114)
ENSG - Equipage, Not Software Generic, generic not available	Generic not available (ENSG)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ICNV - Input Command Not Valid, Interface Standard does not match	modifier consistent with given port's interface standard	"OPR-LPBK: Operate facility loopback" (p. 3-2044); "RLS-LPBK: Release loopback (facility)" (p. 3-2198)

Error code	Meaning (validation rule)	Related to command
ICNV - Input, Command Not Valid	Input failure (ICNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ICNV - Input, Command Not Valid	Login on established session	"ACT-USER: Activate user" (p. 3-24)
ICNV - Input, Command Not Valid, directory/URL must be specified	Directory/URL not specified	"COPY-RFILE: Copy remote file" (p. 3-106)
ICNV - Input, Command Not Valid, directory/URL must be specified	Directory/URL not specified for backup	"COPY-RFILE: Copy remote file" (p. 3-106)
ICNV - Input, Command Not Valid, directory/URL must be specified	Directory/URL not specified for database download	"COPY-RFILE: Copy remote file" (p. 3-106)
ICNV - Input, Command Not Valid, directory/URL must be specified	Directory/URL not specified for restoration	"COPY-RFILE: Copy remote file" (p. 3-106)
ICNV - Input, Command Not Valid, extra AID	Too much AID parameters (ICNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ICNV - Input, Command Not Valid, file must be specified	Directory/URL not specified	"CPY-MEM: Copy memory" (p. 3-114)
ICNV - Input, Command Not Valid, FTTD must be specified	FT-TD device not specified	"COPY-RFILE: Copy remote file" (p. 3-106)
ICNV - Input, Command Not Valid, inmbst not NA	Correct state of tributary for being added (Source)	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against istd	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191) ; "RTRV-VC3: Retrieve VC3 tributary" (p. 3-3069)
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against istd of related port	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against istd of related VCG	"RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109)
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against ptfstd	"RTRV-PTF: Retrieve tributary with path termination function" (p. 3-2800)
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against ptfstd of related tributary	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)

Error code	Meaning (validation rule)	Related to command
ICNV - Input, Command Not Valid, Interface Standard does not match	Check CCB against ptype and istd	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111); "RTRV-STM16: Retrieve STM16 port" (p. 3-2899); "RTRV-STM16T: Retrieve STM16T port" (p. 3-2917)
ICNV - Input, Command Not Valid, Interface Type does not match	Check CCB against ptype and eif	"RTRV-T3: Retrieve DS3 port" (p. 3-2968)
ICNV - Input, Command Not Valid, missing TID	Input failure for omitted TID	"RTRV-HDR: Retrieve header" (p. 3-2417)
ICNV - Input, Command Not Valid, outmbst not NA	Correct state of tributary for being added (Sink)	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
ICNV - Input, Command Not Valid, Protocol Failure in Receive Direction	LCAS failure of protocol present in sink direction	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
ICNV - Input, Command Not Valid, Protocol Failure in Transmit Direction	LCAS failure of protocol present in source direction	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
ICNV - Input, Command Not Valid, required AID missing	Missing AID (ICNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
ICNV - Input, Command Not Valid, missing CTAG	Missing CTAG (ICNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IDNC - Input Data Not Consistent, cross-connection-existence not consistent with cmdmde-value	cmdmde matches cross-connection non-/existence	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
IDNC - Input Data Not Consistent, Inconsistent loopback type w.r.t. aid	Valid loopback type	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
IDNC - Input Data Not Consistent, modifier is inconsistent with existing port rate	modifier consistent with given port-rate	"OPR-LPBK: Operate facility loopback" (p. 3-2044); "RLS-LPBK: Release loopback (facility)" (p. 3-2198)

Error code	Meaning (validation rule)	Related to command
`IDNC - Input Data Not Consistent, Rate is inconsistent with existing cross-connection`	Valid rate	"DLT-CRS: Delete cross-connection" (p. 3-145); "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035); "RLS-LPBK: Release loopback (cross-connection)" (p. 3-2195)
`IDNC - Input Data Not Consistent, Specified Tributary does not support the indicated rate`	Tributary provisioning and rate consistency	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`IDNC - Input Data Not Consistent, Tributary boundary is inconsistent with the indicated rate`	AID boundary and rate consistency	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`IDNC - Input Data Not Consistent, VGC-trib as input-trib in pathprotected crosconnection`	No VCG-trib as in_aid (for path-protected cross-connection)	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`IDNC - Input, Data Not Consistent`	all or no traffic for SCN node	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`IDNC - Input, Data Not Consistent`	can't enable isis_lanstat without enabling osi_lanstat	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`IDNC - Input, Data Not Consistent`	disabling of superusers	"ED-USER-SECU: Edit user security" (p. 3-1328)
`IDNC - Input, Data Not Consistent`	Invalid Input Parameter Value Combination (IDNC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IDNC - Input, Data Not Consistent`	l4ajsys must be supplied	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`IDNC - Input, Data Not Consistent, a new trace value must also be supplied for a trace format change`	Supply of outgoing J0 trace	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNC - Input, Data Not Consistent, a new trc must also be supplied for a path trace mode change`	supply of transmitted path trace	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data Not Consistent, 'ALL' qualifier for port aid not allowed	'ALL' qualifier used for wrong parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111); "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNC - Input, Data Not Consistent, Assigned Node not allowed for Auto Encapsulation	Only osinode1 supports Auto Encapsulation	"ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
IDNC - Input, Data Not Consistent, Connections between BLSR/MS-SPRing working and protection tributaries are not allowed	No connections between BLSR/MS-SPRing working and protection tribs	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, conversion cross-connection not supported by that pack	Pack restrictions for conversion cross-connection sources	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, Corresponding loca/ locz information is missing	Corresponding loca/loca2/locz information is missing	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, default profile	deleting the system default profile	"DLT-TCA-PROF: Delete TCA profile" (p. 3-259)
IDNC - Input, Data Not Consistent, Default VLAN ID is already assigned for this System	Default VLAN ID is already assigned for this system	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNC - Input, Data Not Consistent, Flow Control settings do not match to existing autonegotiation mode	GE10PL1 Pack specific validation rule: No auto flow control settings if autonegotiation disabled	"ED-EPORT: Edit Ethernet port" (p. 3-329)
IDNC - Input, Data Not Consistent, Flow Control settings do not match to existing autonegotiation mode	GE10PL1 Pack specific validation rule: No fix flow control settings if autonegotiation enabled	"ED-EPORT: Edit Ethernet port" (p. 3-329)
IDNC - Input, Data Not Consistent, Inconsistent loca/ locz information	Consistent loca/loca2/locz for all low-order cross-connections using the same HO PTF at an MS-SPRing port	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, Inconsistent loca/ locz information	Consistent loca/locz for all conversion cross-connections using the same HO PTF at an MS-SPRing port	"ENT-CRS: Enter cross-connection" (p. 3-1464)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data Not Consistent, IP address of level 1 Routing Controller already in use	SCN IP address already in use	"ACT-L1ROUTING" (p. 3-15)
IDNC - Input, Data Not Consistent, It is not allowed to specify loca/locz for EXTRA TRAFFIC	loca/locz information only retrievable for EXTRA TRAFFIC	"ED-CRS: Edit cross-connection" (p. 3-292)
IDNC - Input, Data Not Consistent, It is not allowed to specify loca/locz for EXTRA TRAFFIC	newloca not provisionable for Extra Traffic	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data Not Consistent, level 1 Routing Controller already in place	level 1 Routing controller in place	"ACT-L1ROUTING" (p. 3-15)
IDNC - Input, Data Not Consistent, LO tributary not supported by that pack	Pack restrictions for LO substructure creation	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
IDNC - Input, Data Not Consistent, LO tributary not supported by that pack	Pack restrictions for LO tributaries under test	"CONN-TACC: Connect test access" (p. 3-94)
IDNC - Input, Data Not Consistent, LO tributary not supported by that pack	Pack restrictions for LOXC cross-connection sources	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, modifier is inconsistent with existing port rate	modifier consistent with given port-rate	"ED-EPORT: Edit Ethernet port" (p. 3-329); "ED-VCG: Edit virtual concatenation group" (p. 3-1354); "RTRV-EPORT: Retrieve Ethernet port" (p. 3-2358)
IDNC - Input, Data Not Consistent, modifier is inconsistent with existing port rate	modifier consistent with given port-rate;	"RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)
IDNC - Input, Data Not Consistent, modifier is inconsistent with the related circuit pack type	modifier consistent with the related circuit pack type	"ED-EPORT: Edit Ethernet port" (p. 3-329); "OPR-LPBK: Operate facility loopback" (p. 3-2044); "RLS-LPBK: Release loopback (facility)" (p. 3-2198); "RTRV-EPORT: Retrieve Ethernet port" (p. 3-2358)
IDNC - Input, Data Not Consistent, modifier is required for this circuit pack type	modifier mandatory for the GE10PL1 and ES64U circuit pack	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNC - Input, Data Not Consistent, modifier is required for this circuit pack type	modifier mandatory for the GE10PL1 and ES64U circuit pack;	"RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data Not Consistent, MSP 1+1 optimized not supported for SONET mode	No MSP 1+1 optimized protection for SONET	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data Not Consistent, new name for default profile	rename default profile name	"ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
IDNC - Input, Data Not Consistent, new name for default profile	Rename of default ASAP	"ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
IDNC - Input, Data Not Consistent, No low-order path protection with input tributaries from 2 fiber MS-SPRing/BLSR protection	No low-order path protection input tributaries from the protection bandwith of 2-fiber MS-SPRing/BLSR protection	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data Not Consistent, No low-order path protection with input tributaries from 4-fiber BLSR/MS-SPRing ports	No low-order connection with 4 fiber BLSR/MS-SPRing ports	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data Not Consistent, No low-order path protection with input tributaries from BLSR/MS-SPRing ports	No low-order path protection with input tributaries from BLSR/MS-SPRing ports	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, No low-order path protection with input tributaries from protection bandwidth of 2-fiber MS-SPRing/BLSR protection	No low-order path protection with input tributaries from the protection bandwidth of 2-fiber MS-SPRing/BLSR protection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, no OPT2G5-trib as input-trib in pathprotected crossconnection	No OPT2G5 related trib as in_aid (for pathprotected cross-connection)	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, No trib of MS-SPRing/ BLSR port allowed for conversion cross-connection	No tributary of an MS-SPRing/BLSR port for conversion cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, NSA severity is higher than SA severity	Inconsistent severity level	"ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
IDNC - Input, Data Not Consistent, only one of dccstat, scn_dccstat, uni_dccstat, or enni_ dccstat allowed	Single occurence of dccstat, scn_dccstat, uni_dccstat, or enni_dccstat parameter	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNC - Input, Data Not Consistent, pathprotected conversion cross-connections not allowed	No path-protected conversion cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data Not Consistent, port_aid1 and port_aid2 not different	Different port AIDs	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
IDNC - Input, Data Not Consistent, primary/secondary test access tributary not supported by that pack	Pack restrictions for primary and secondary HO test access tributary	"CONN-TACC: Connect test access" (p. 3-94)
IDNC - Input, Data Not Consistent, primary/secondary test access tributary not supported by that pack	Pack restrictions for primary and secondary LO test access tributary	"CONN-TACC: Connect test access" (p. 3-94)
IDNC - Input, Data Not Consistent, raw_port and lv_port must be unequal	check that raw_port and lv_port are unequal	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
IDNC - Input, Data Not Consistent, RC PC SCN address must be different from SC PC SCN address	SCN IP address already in use 2	"ACT-L1ROUTING" (p. 3-15)
IDNC - Input, Data Not Consistent, SCN not enabled	Check of SCN availability	"ACT-L1ROUTING" (p. 3-15)
IDNC - Input, Data Not Consistent, sigPCIP address and RC IP address must not be equal	SC IP Address and RC IP address must be different	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
IDNC - Input, Data Not Consistent, sigPCIP Address and sigPCID must be supplied together	sigPCIP_Address and sigPCID must be supplied together.	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
IDNC - Input, Data Not Consistent, Specified Tributary cannot be used for LOXC connections	Restrictions for LOXC cross-connection sources	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data Not Consistent, Specified Tributary does not support the indicated rate	HO tributary rate consistency	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
IDNC - Input, Data Not Consistent, Specified Tributary does not support the indicated rate	Restrictions for LO SONET test access rates	"CONN-TACC: Connect test access" (p. 3-94)
IDNC - Input, Data Not Consistent, Specified Tributary does not support the indicated rate	Restrictions for LOXC cross-connection rates	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, Specified Tributary does not support the indicated rate	Tributary provisioning and rate consistency	"CONN-TACC: Connect test access" (p. 3-94)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data Not Consistent, strcout length must match the given outgoing J0 trace mode	transmitted path trace length	"ED-EC1: Edit EC1 port" (p. 3-317)
IDNC - Input, Data Not Consistent, This MSP protection type not supported for a port with the indicated rate	No MSP/APS 1+1 for bitrate STM-256/OC-768	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data Not Consistent, trc length must match the given path trace mode	transmitted path trace length	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)
IDNC - Input, Data Not Consistent, Tributaries must not be on ports of portClass ENNI	LO Substructure creation not allowed on ENNI ports	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
IDNC - Input, Data Not Consistent, Tributaries must not be on ports of portClass INNI	(ONNS specific): LO Tributaries must not belong to ports of portClass INNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data Not Consistent, Tributaries must not be on ports of portClass INNI	LO Substructure creation not allowed on INNI ports	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
IDNC - Input, Data Not Consistent, Tributaries must not be on ports of portClass UNI	LO Substructure creation not allowed on UNI ports	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
IDNC - Input, Data Not Consistent, Tributary boundary is inconsistent with the indicated rate	LO-AID boundary and rate consistency	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
IDNC - Input, Data Not Consistent, UNI IP address, sigPCIP address, and ASTN IP address must not be equal	UNI IP address, SC IP Address and SCN IP address must be different.	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
IDNC - Input, Data Not Consistent, value change not allowed for profile Default0	value change of default profile Default0	"ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
IDNC - Input, Data Not Consistent, VCG-trib as input-trib in pathprotected crossconnection	Tributaries must not be VCG tributaries	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
IDNC - Input, Data, Not Consistent	Can't assign same IP address to several LANs	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
IDNC - Input, Data, Not Consistent	Can't enable ip_lanstat and/or osi_lanstat when lanstat is disabled	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
IDNC - Input, Data, Not Consistent	Can't enable lanstat when ip_lanstat and osi_lanstat are DISABLED	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data, Not Consistent, 2f-BSLR rate exceeds half of the port rate	Rate must not exceed half of the port rate (2f-BSLR only)	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data, Not Consistent, a new trc must also be supplied for a path trace mode change	supply of transmitted path trace	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
IDNC - Input, Data, Not Consistent, Bridging rate is not compliant with existing leg	Both legs must have same rate	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data, Not Consistent, default profile	pfname is default	"DLT-ASAP-PROF: Delete alarm severity assignment profile" (p. 3-134)
IDNC - Input, Data, Not Consistent, MSP 1+1 Optimized is always non-revertive	MSP type 1+1_optimized is never revertive	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, MSP Protection Type 1x1 is always revertive	MSP type 1:1 is never non-revertive	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, no allowed topology conversion	Topology conversion command must be consistent	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data, Not Consistent, No MSP protection supported for SONET mode	No MSP 1:1 protection for SONET	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, not own uid	not the own uid specified	"CANC-USER: Cancel user" (p. 3-60)
IDNC - Input, Data, Not Consistent, Optimized exists only with 1+1 Protection Type	only MSP 1+1 can be optimized	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, SNC Type not compliant with existing pathprotected crossconnection	SNC type compliance	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data, Not Consistent, SNC Type not compliant with existing path-protected cross-connection	SNC Type Compliance	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data, Not Consistent, Specified new Tributary does not support the indicated rate	New tributary rate validity	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - Input, Data, Not Consistent, Specified Tributary does not support the indicated rate	Tributary provisioning and rate consistency	"ENT-CRS: Enter cross-connection" (p. 3-1464)

Error code	Meaning (validation rule)	Related to command
IDNC - Input, Data, Not Consistent, strcout length must match the given outgoing J0 trace mode	transmitted path trace length	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNC - Input, Data, Not Consistent, This MSP protection type not supported for a port with the indicated rate	No MSP 1:1 protection for bitrate STM-256	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, This MSP protection type not supported for a port with the indicated rate	No MSP 1+1 optimized for bitrate STM-256	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNC - Input, Data, Not Consistent, trc length must match the given path trace mode	transmitted path trace length	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
IDNC - Input, Data, Not Consistent, Tributary boundary is inconsistent with the indicated rate	AID boundary and rate consistency	"CONN-TACC: Connect test access" (p. 3-94); "ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNC - Input, Data, Not Consistent, VCG-trib as input-trib in pathprotected crossconnection	No VCG-trib as in_aid (for pathprotected cross-connection)	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC - SARB - Status, All Resources Busy, LOXC capacity exhausted	Spare capacity for HO cross-connection legs on LOXC	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IDNC- Input, Data Not Consistent, level2 not allowed for scnnode	level2 parameter must not be present when osinode = scnnode	"RTRV-MAP-NETWORK: Retrieve network map" (p. 3-2526)
IDNC-Input, Data Not Consistent, Default VLAN ID is not an allowed member of valid VLAN list	BCMD mode - Default VLAN ID is no allowed member of valid VLAN list for this System	"ED-EPORT: Edit Ethernet port" (p. 3-329)
IDNC-Input, Data, Not Consistent, Default VLAN ID is already assigned for this System	Default VLAN ID is already assigned for this system	"ED-EPORT: Edit Ethernet port" (p. 3-329)
IDNV - Input Data Not Vaild, Both input tributaries are identical	path-protected cross-connection has unique input AIDs	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNV - Input Data Not Valid, ASAP profile does not exist	pfname does not exist	"SET-ATTR-ENV: Set attribute environment" (p. 3-3137)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input Data Not Valid, invalid atagseg`	valid atagseq	"RTRV-AO: Retrieve autonomous output" (p. 3-2241)
`IDNV - Input Data Not Valid, Invalid loopback for EPort`	For ports on GEn packs only far-side loopback allowed	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
`IDNV - Input Data Not Valid, osinode cannot be changed for LAN1`	only osinode1 for LAN 1	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`IDNV - Input Data Not Valid, Parameter(s) present which are not allowed`	Only those parameter present which are needed	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`IDNV - Input, Data Not Vaild, Both input tributaries are identical`	path-protected cross-connection has unique input AIDs	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`IDNV - Input, Data Not Valid`	Any Input Parameter not valid (IDNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IDNV - Input, Data Not Valid`	Check for that either dccstat, scn_dccstat, uni_dccstat, or enni_dccstat is present	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`IDNV - Input, Data Not Valid`	Disabled User ID	"ACT-USER: Activate user" (p. 3-24)
`IDNV - Input, Data Not Valid`	Expired password and unsuccessful change of password	"ACT-USER: Activate user" (p. 3-24)
`IDNV - Input, Data Not Valid`	Input parameters not specified	"COPY-RFILE: Copy remote file" (p. 3-106); "CPY-MEM: Copy memory" (p. 3-114)
`IDNV - Input, Data Not Valid`	Invalid uid and/or pid	"ACT-USER: Activate user" (p. 3-24)
`IDNV - Input, Data Not Valid`	Parameter nvmname must be specified	"ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547); "ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
`IDNV - Input, Data Not Valid`	Presence of spec_block parameter	"ED-CRS: Edit cross-connection" (p. 3-292)
`IDNV - Input, Data Not Valid, a loopback is existing`	No interface standard change on GE10PL1 pack with existing loopbacks	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, additional parameters required	Additional parameters required when setting ndetmd to MAN	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, AIS mode not valid	Output type not supported for AIS mode	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, all tributaries must be different	All tributaries must be different	"CONN-TACC: Connect test access" (p. 3-94)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, ASAP profile does not exist`	ASAP profile exists	“ED-EC1: Edit EC1 port” (p. 3-317); “ED-EPORT: Edit Ethernet port” (p. 3-329); “ED-LONIM: Edit lower-order non-intrusive monitor” (p. 3-411); “ED-LOPTF: Edit lower-order path termination function” (p. 3-427); “ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585); “ED-OC48T: Edit OC48 transparent port” (p. 3-648); “ED-PTF: Edit tributary with Path Termination Function (PTF)” (p. 3-830); “ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM16T: Edit STM16T port” (p. 3-1097); “ED-STM1E: Edit STM1E port” (p. 3-870); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111); “ED-STS1: Edit STS1 tributary” (p. 3-1177); “ED-T3: Edit DS3 port” (p. 3-1202); “ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card” (p. 3-1306); “ED-VC3: Edit VC3 tributary” (p. 3-1339); “ED-VCG: Edit virtual concatenation group” (p. 3-1354); “ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application” (p. 3-1405); “ED-VCGTRIB: Edit virtual concatenation group tributary” (p. 3-1416)
`IDNV - Input, Data Not Valid, ASAP profile does not exist`	check for ASAP profile existence	“ED-EQPT (system): Edit equipment (system)” (p. 3-391)
`IDNV - Input, Data Not Valid, ASAP profile does not exist`	Check for ASAP profile existence	“ED-EQPT (shelf): Edit equipment (shelf)” (p. 3-365); “ED-EQPT (slot): Edit equipment (slot)” (p. 3-370); “ED-EQPT (socket): Edit equipment” (p. 3-379)
`IDNV - Input, Data Not Valid, ASAP profile does not exist`	Validate ASAP profile	“ED-EQPT (CP): Edit equipment (circuit pack)” (p. 3-349)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, Assigned bandwidth exceeds half port’s bandwith	Enough bandwidth reserved for 2F protection group	“ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC48: Edit OC48 port” (p. 3-585); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM64: Edit STM64 port” (p. 3-1111)
IDNV - Input, Data Not Valid, Automatic Squelch Map Calculation only allowed with Automatic Ring Discovery	Automatic squelch map calculation only with automatic ring discovery	“ED-PROTN-GRP: Edit protection group” (p. 3-803); “ENT-PROTN-GRP: Enter protection group” (p. 3-1745)
IDNV - Input, Data Not Valid, cable equalization not supported	Cable equalization not supported	“SET-SYNCN: Set synchronization” (p. 3-3158)
IDNV - Input, Data Not Valid, can’t change l3domain for osinode1	l3domain can not be changed for osinode1	“ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack” (p. 3-1312)
IDNV - Input, Data Not Valid, change of worker port not allowed	Unallowed modification of worker ports	“ED-PROTN-GRP: Edit protection group” (p. 3-803)
IDNV - Input, Data Not Valid, Client Node ID must differ from Network Node ID	Check of UNI parameters	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585); “ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM1E: Edit STM1E port” (p. 3-870); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111); “ED-VCG: Edit virtual concatenation group” (p. 3-1354)
IDNV - Input, Data Not Valid, Combination of Modifier/ Parameters not allowed	Parameter Combinations	“DLT-ENNIMANROUTE: Delete manual route from the E-NNI node” (p. 3-164); “DLT-ENNIMANROUTE notification” (p. 3-168); “ENT-ENNIMANROUTE: Enter manual route to the E-NNI node” (p. 3-1520); “ENT-ENNIMANROUTE notification” (p. 3-1528)
IDNV - Input, Data Not Valid, Command not supported for test access rate LOVT1.5	Not Supported for Low-Order Test Access Rate	“CHG-ACCMD: Change access mode” (p. 3-65)
IDNV - Input, Data Not Valid, Consistent TNA parameter set required	(ONNS specific): Complete and Consistent Setting of TNA Parameter required when configured to EDGE VCGs	“ED-VCG: Edit virtual concatenation group” (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, Consistent TNA parameter set required	(ONNS specific): Consistent Parameters setting required when removing TNA information from EDGE VCGs	“ED-VCG: Edit virtual concatenation group” (p. 3-1354)
IDNV - Input, Data Not Valid, Consistent TNA parameter set required	Complete and Consistent Setting of TNA Parameter required when configured to EDGE ports	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585); “ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM1E: Edit STM1E port” (p. 3-870); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111)
IDNV - Input, Data Not Valid, Consistent TNA parameter set required	Consistent Parameters setting required when removing TNA information from EDGE ports	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585); “ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM1E: Edit STM1E port” (p. 3-870); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111)
IDNV - Input, Data Not Valid, constraint l3radesnode	l3radesnode value and l3isisnl value	“ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack” (p. 3-1312); “ENT-ULSDCC-L3: Create a further OSI node” (p. 3-1951)
IDNV - Input, Data Not Valid, Conversion cross-connections not supported with current LO interface standard	Conversion cross-connections only with LOXC interface standard SDH	“ENT-CRS: Enter cross-connection” (p. 3-1464)
IDNV - Input, Data Not Valid, Cross-connections still existing	No interface standard change on GE10PL1 pack with existing cross-connection	“ED-EQPT (CP): Edit equipment (circuit pack)” (p. 3-349)
IDNV - Input, Data Not Valid, DCC on Transparent Ports not allowed	No DCC on transparent ports	“ENT-CRS-FECOM: Enter transparent DCC cross-connection” (p. 3-1510); “ENT-FECOM: Enter DCC channel related parameters” (p. 3-1565)
IDNV - Input, Data Not Valid, DCC type SECTION not allowed for SCN	(SCN specific): RS-DCC not allowed for SCN	“ENT-FECOM: Enter DCC channel related parameters” (p. 3-1565)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, different number of values in rnids and rtids	Consistent rnids and rtids lists	"ED-MAP-RING: Edit map ring" (p. 3-436)
IDNV - Input, Data Not Valid, discrepancy between entered values and current value of spec-block parameters	ohbyte and intfctype values correct (when dis-associating)	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
IDNV - Input, Data Not Valid, DS3 Ports not allowed as line timing reference	No DS3 Port as line timing reference	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, duplicate IP Map entry	Check for duplicate entries	"ENT-IP-MAP: Enter TCP/IP map" (p. 3-1594)
IDNV - Input, Data Not Valid, duplicate neighbor port information	No setting of duplicate neighborASTNNodeID and neighborPortID parameter combination	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, duplicate ring node identifiers	No duplicate Ring Node Identifiers	"ED-MAP-RING: Edit map ring" (p. 3-436)
IDNV - Input, Data Not Valid, duplicate ring node target identifiers	No duplicate Ring Node Target Identifiers	"ED-MAP-RING: Edit map ring" (p. 3-436)
IDNV - Input, Data Not Valid, duplicate ring node TID	No duplicate ring node TIDs	"ED-FETID: Edit far end target identifier" (p. 3-405)
IDNV - Input, Data Not Valid, duplicate UID	Entering user's own or existing uid.	"ENT-USER-SECU: Enter user security" (p. 3-1970)
IDNV - Input, Data Not Valid, duplicate UID	newuid has the same value of an existing uid	"ED-USER-SECU: Edit user security" (p. 3-1328)
IDNV - Input, Data Not Valid, E and F tributaries not cross-connected	E tributary and F tributary not cross-connected	"CONN-TACC: Connect test access" (p. 3-94)
IDNV - Input, Data Not Valid, EC1 Ports not allowed as line timing reference	No EC1 Port as line timing reference	"SET-SYNCN: Set synchronization" (p. 3-3158)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, ENNI Port Parameters must be different to their DEFAULT values`	Check of ENNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNV - Input, Data Not Valid, equal l3domain`	l3domain value must be different for current node and l3radesnode	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
`IDNV - Input, Data Not Valid, equal node IDs`	l3radesnode value must be different from osinode	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312); "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
`IDNV - Input, Data Not Valid, forced DUS not supported for this format`	Format not supported for forced DNU/DUS	"SET-SYNCN: Set synchronization" (p. 3-3158)
`IDNV - Input, Data Not Valid, given TNA already exists`	(ONNS specific): TNA parameters on SDH/SONET layer already assigned on VCAT layer;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, incompatible screen, uidclass parameters`	Compatibility screen, uidclass parameter	"ENT-USER-SECU: Enter user security" (p. 3-1970); "ED-USER-SECU: Edit user security" (p. 3-1328)
`IDNV - Input, Data Not Valid, incomplete switch command`	complete switch command	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
`IDNV - Input, Data Not Valid, inconsistency l3domain with l3radesnode`	consistency l3domain with l3radesnode for nodes 2 to 8	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312); "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
`IDNV - Input, Data Not Valid, inconsistent aid and protype`	consistent aid and protype	"RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, inconsistent aid and rid	consistent aid and rid	"DLT-PROTN-GRP: Delete protection group" (p. 3-233); "ED-FETID: Edit far end target identifier" (p. 3-405); "ED-MAP-RING: Edit map ring" (p. 3-436); "ED-PROTN-GRP: Edit protection group" (p. 3-803); "OPR-PROTNSW: Operate protection switch" (p. 3-2057); "RTRV-MAP-RING: Retrieve map ring" (p. 3-2532); "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759); "SET-MAP-RING: Set map ring" (p. 3-3142)
IDNV - Input, Data Not Valid, incorrect date/time format	Incorrect date/time format	"ED-DAT: Edit date and time" (p. 3-312)
IDNV - Input, Data Not Valid, incorrect new_pid	Missing or invalid or syntactically incorrect new_pid	"ED-PID: Edit password identifier" (p. 3-798)
IDNV - Input, Data Not Valid, input parameters are not matching	Input Parameters not matching	"DLT-CONN: Delete network connection" (p. 3-139); "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
IDNV - Input, Data Not Valid, Interface Standard cannot be modified	Not allowed to change istd for VCG ports on a GE10PL1 pack	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, Interface type not settable for specified UserByte-Port	Interface type compliant with port	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
IDNV - Input, Data Not Valid, invalid <parameter_name>	Input Parameter not valid (IDNV)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IDNV - Input, Data Not Valid, invalid callID	Input Parameter callID not valid	"DLT-CONN: Delete network connection" (p. 3-139); "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
IDNV - Input, Data Not Valid, invalid change of LO switch capacity	Only decreasing of LO switch capacity to 0 G allowed	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
IDNV - Input, Data Not Valid, invalid connection ID	Input Parameter connID not valid	"DLT-CONN: Delete network connection" (p. 3-139); "RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
IDNV - Input, Data Not Valid, invalid Destination TNA Address	Input Parameter Destination TNA Address not valid	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, invalid ES64U switch capacity for ES64U in this slot`	ES64U capacity setting not for ES64U in protection functionality;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`IDNV - Input, Data Not Valid, invalid LO switch capacity for LOXC in this slot`	LO switch capacity setting only for LOXC in worker slots	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`IDNV - Input, Data Not Valid, invalid LO switch capacity for the LOXC type`	LO switch capacity allowed for both LOXC in an equipment protection group	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`IDNV - Input, Data Not Valid, invalid LO switch capacity for the LOXC type`	LO switch capacity allowed for LOXC type	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`IDNV - Input, Data Not Valid, Invalid loopback for PTF on LOXC`	No loopbacks for PTF on LOXC allowed	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
`IDNV - Input, Data Not Valid, invalid sc`	Invalid switch command	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
`IDNV - Input, Data Not Valid, invalid Service Level`	Input Parameter Service Level not valid	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
`IDNV - Input, Data Not Valid, invalid slots for worker and protection unit`	Slot restrictions for TransMUX equipment protection group	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`IDNV - Input, Data Not Valid, invalid Source LPID`	Input Parameter Source Logical Port ID not valid	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
`IDNV - Input, Data Not Valid, invalid Source TNA Address`	Input Parameter Source TNA Address not valid	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
`IDNV - Input, Data Not Valid, invalid switch command`	supported switch command	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
`IDNV - Input, Data Not Valid, invalid test access mode`	MONF or MONEF test access mode not allowed on a one-way cross-connection from E to F tributary	"CONN-TACC: Connect test access" (p. 3-94)
`IDNV - Input, Data Not Valid, invalid TNA address`	Input Parameter TNA address not valid	"RTRV-CONN: Retrieve data relevant for network connections" (p. 3-2294)
`IDNV - Input, Data Not Valid, invalid TransMUX switch capacity for TransMUX in this slot`	TransMUX capacity setting not for TransMUX in protection functionality	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`IDNV - Input, Data Not Valid, invalid UID`	invalid uid	"RTRV-USER-SECU: Retrieve user security" (p. 3-3063)
`IDNV - Input, Data Not Valid, invalid UID`	Invalid uid	"DLT-USER-SECU: Delete user security" (p. 3-272); "ED-USER-SECU: Edit user security" (p. 3-1328)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, invalid UPC	Invalid upc	"ED-USER-SECU: Edit user security" (p. 3-1328); "ENT-USER-SECU: Enter user security" (p. 3-1970)
IDNV - Input, Data Not Valid, Invalid Wildcarding for wrk_aid	Invalid wildcarding	"RTRV-OUC: Retrieve ONNS to UPSR/SNCP constructs" (p. 3-2690)
IDNV - Input, Data Not Valid, IP Address already used	dcn_ipad must be unique in the system ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
IDNV - Input, Data Not Valid, IP Address already used	IP address already used for an active ES64U pack;	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
IDNV - Input, Data Not Valid, l3prefix and l3radesnode inconsistent	consistency of l3radesnode with l3raprefix	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312); "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
IDNV - Input, Data Not Valid, l4ajsys/osinode not unique	l4ajsys per osinode must be different	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
IDNV - Input, Data Not Valid, Labels and Bandwidth are not matching	Labels and bandwidth not matching	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
IDNV - Input, Data Not Valid, LAN1 not allowed for ENNI	Not allowed to use LAN1 for E-NNI	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528)
IDNV - Input, Data Not Valid, LAN1 not allowed for UNI	Not allowed to use LAN1 for UNI	"DLT-MANROUTE-LAN: Delete manual route at LAN port" (p. 3-203); "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610); "RTRV-MANROUTE-LAN: Retrieve manual route at LAN port" (p. 3-2516)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, local Link ID value must be unique`	Unique setting of localLinkID	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): localLPID already assigned to VCG port as localLPIDVCAT	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): localLPIDVCAT already assigned to other EDGE or SONET/SDH UNI port	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): localLPIDVCAT already assigned to other VCG port	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): Unique setting of localLPID	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): Unique setting of localLPID wrt. localLinkID	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	(ONNS specific): Unique setting of localLPIDVCAT wrt. localLinkID	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	localLPID already assigned to VCG port at VCAT layer	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNV - Input, Data Not Valid, local LPID value must be unique`	Unique setting of localLinkID wrt. localLPID	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, local LPID value must be unique	Unique setting of localLPID	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, local LPID value must be unique	Unique setting of localLPID wrt. localLinkID	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, Low-order cross-connections of this interface standard not supported	Entities must fit LOXC interface standard	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191) ; "ENT-CRS: Enter cross-connection" (p. 3-1464)
IDNV - Input, Data Not Valid, Low-order test access sessions of this interface standard not supported	Entities must fit LOXC interface standard	"CONN-TACC: Connect test access" (p. 3-94)
IDNV - Input, Data Not Valid, Manual route entry already exists	No duplicate Manual Route entries	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528); "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610)
IDNV - Input, Data Not Valid, Manual route entry does not exist	Manual MCN IP Route entry must exist	"DLT-MANROUTE-MCN: Delete one manual MCN IP route entry" (p. 3-207)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, Manual route entry does not exist`	Manual Route entry must exist	"DLT-ENNIMANROUTE: Delete manual route from the E-NNI node" (p. 3-164); "DLT-MANROUTE-LAN: Delete manual route at LAN port" (p. 3-203)
`IDNV - Input, Data Not Valid, Min. and max. NN XC rates incompatible`	No setting of minNNXCRate to values larger than value of maxNNXCRate and vice versa	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNV - Input, Data Not Valid, missing F tributary`	Missing F tributary	"CONN-TACC: Connect test access" (p. 3-94)
`IDNV - Input, Data Not Valid, neighboring RC ID/IP address used by own RC`	Neighboring RC must differ from own RC	"ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)
`IDNV - Input, Data Not Valid, newuid for non superuser`	newuid defined for non superuser uid	"ED-USER-SECU: Edit user security" (p. 3-1328)
`IDNV - Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`	No duplicate Manual MCN IP Route entries	"ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
`IDNV - Input, Data Not Valid, Next Hop IP Address already assigned to other LAN`	No more than one LAN connected to same next hop	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528); "ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
`IDNV - Input, Data Not Valid, No Access via AID located on INNI port`	Access via tributary on INNI port	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "OPR-PROTNSW: Operate protection switch" (p. 3-2057); "RLS-PROTNSW: Release protection switch" (p. 3-2203)
`IDNV - Input, Data Not Valid, no aid range in combination with rid`	aid range and rid	"RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, no Auto. Squelch Map Calculation	no automatic squelch map calculation	"RTRV-CKT: Retrieve circuit" (p. 3-2274)
IDNV - Input, Data Not Valid, no BLSR / MS SPRING tributary	no BLSR / MS-SPRING tributary	"RTRV-CKT: Retrieve circuit" (p. 3-2274)
IDNV - Input, Data Not Valid, No mixing of EDGE and UNI ports under one TNA	TNA parameter disjointness between EDGE and UNI;	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, No path protection group parameters allowed for directionality UNIDIR-ADD	No path protection group parameters must be given for Setup of a unidirectional-add OUC	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
IDNV - Input, Data Not Valid, no regen loop with SETG as timing output	Regenerator loops with SETG as timing output	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, No support of TNA parameter set for UNI VCGs	Check of SONET/SDH layer parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, No support of VCAT TNA parameter set for EDGE VCGs	(ONNS specific): No Support of VCAT related TNA Parameters for EDGE VCGs;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, No Userbyte access at the specified port	No Userbyte access at EC1 ports on EP51 pack	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
IDNV - Input, Data Not Valid, No wildcarding for access via input tribs	No wildcarding on access via aid and aid1	"RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
IDNV - Input, Data Not Valid, node itself is not a valid neighbor	No setting of neighborASTNNodeID or neighborNodeID to corresponding IDs of local node	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, not possible for NNI port to assign both channels to MCN	(SCN specific): Prevent that both DCCs of a port are assigned to the MCN, when portClass = INNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, not possible to assign both channels to E-NNI	(UNI/E-NNI specific): Prevent that both DCCs of a port are assigned to E-NNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, not possible to assign both channels to SCN	(SCN specific): Prevent that both DCCs of a port are assigned to the SCN	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, not possible to assign both channels to UNI	(UNI/E-NNI specific): Prevent that both DCCs of a port are assigned to UNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, NSAP Area Addres matches with RA-Prefix	Address area value - RA-prefix value	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
IDNV - Input, Data Not Valid, NSAP Area Address matches with RA-Prefix	Address area value - RA-Prefix value	"ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
IDNV - Input, Data Not Valid, old TID not in in the list of ring node TIDs	Old TID not in ring map	"ED-FETID: Edit far end target identifier" (p. 3-405)
IDNV - Input, Data Not Valid, only a single node in separate domain supported	only one single node in a separate domain supported	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312); "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
IDNV - Input, Data Not Valid, optical wavelength not in optical window	Optical wavlength must be in the optical window	"ED-OTU2: Edit OTU2 port" (p. 3-730)
IDNV - Input, Data Not Valid, Parameter inputsig is mandatory	Parameter inputsig is mandatory	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)
IDNV - Input, Data Not Valid, parameter inputsig must be set individually	No other spec_block parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, parameter not required	Parameter not supported for modifier	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
IDNV - Input, Data Not Valid, Parameter not supported	Change of FEC parameters only if supported by the current circuit pack	"ED-OC192: Edit OC192 port" (p. 3-663); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, Parameter not supported	Enabling of Optical Channel only if supported by the current circuit pack	"ED-OC192: Edit OC192 port" (p. 3-663); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, Parameter vcat is mandatory	Parameter vcat is mandatory	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, parameter vcat must be set individually	No other spec_block parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, Parameter(s) present which are not allowed	Parameters not supported on GE1 pack	"ED-EPORT: Edit Ethernet port" (p. 3-329)
IDNV - Input, Data Not Valid, Port Class not allowed for 10 GE ports	Ethernet Port Class EDGE not allowed for 10 GE ports	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, Port Class not allowed for 10 GE ports	Ethernet Port Class UNI not allowed for 10 GE ports	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, portClass must be ENNI	(UNI/E-NNI specific): Check consistency of input parameters with portClass	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, portClass must be INNI	(SCN specific): Check consistency of input parameters with portClass I	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, portClass must be UNI	(UNI/E-NNI specific): Check consistency of input parameters with portClass	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, ppgname not allowed if aid1 is given	ppgname and aid1	"RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
IDNV - Input, Data Not Valid, ppgname not supported for this transmission rate	ppgname and low order path protections	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "OPR-PROTNSW: Operate protection switch" (p. 3-2057); "RLS-PROTNSW: Release protection switch" (p. 3-2203); "RTRV-PROTN-GRP: Retrieve protection group" (p. 3-2759)
IDNV - Input, Data Not Valid, profile name does not exist	profile name does not exist	"DLT-TCA-PROF: Delete TCA profile" (p. 3-259); "ED-TCA-PROF: Edit TCA profile" (p. 3-1214)
IDNV - Input, Data Not Valid, provisioning of lapdmode and lapdrl not possible	(SCN specific): Invalid input parameters	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
IDNV - Input, Data Not Valid, RA-Prefix matches with the NSAP Area Address	RA-prefix value - Address area value	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312); "ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, RC neighbor already provisioned	RC must not be in Neighboring List	"ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)
IDNV - Input, Data Not Valid, RC node not created	RC must be created	"DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers" (p. 3-247); "ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)
IDNV - Input, Data Not Valid, RC not created	RC must be created	"ED-RC-GENERAL: Edit general routing controller settings" (p. 3-853); "RTRV-RC-GENERAL: Retrieve general routing controller settings" (p. 3-2814); "RTRV-RC-NEIGHBOR: Retrieve all neighboring routing controllers" (p. 3-2816)
IDNV - Input, Data Not Valid, RC unknown	RC must be in Neighboring List	"DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers" (p. 3-247)
IDNV - Input, Data Not Valid, related portClass not EDGE or port not in BLSR/MS-SPRING group	(ONNS specific): loca, locz changes only allowed when portClass = EDGE and part of a BLSR/MS-SPRING group	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
IDNV - Input, Data Not Valid, rid is already used by another protection group	unique rid	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNV - Input, Data Not Valid, Sa bit location not supported	Input / Output format not supported for Sa bit settings	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, screen value was not set by security administrator	screen value not allowed for uid	"ED-USER: Edit user" (p. 3-1323)
IDNV - Input, Data Not Valid, sid is already used by another BLSR/ MS-SPRing Node	Uniqueness of TID within Ringmaps	"SET-SID: Set SID" (p. 3-3154)
IDNV - Input, Data Not Valid, side information is not complete	complete side information	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
IDNV - Input, Data Not Valid, Slot still provisioned	Slot must not be provisioned	"ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
IDNV - Input, Data Not Valid, Socket still provisioned	Socket must not be provisioned	"ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
IDNV - Input, Data Not Valid, SSM not supported for provisioned QL	SSM not supported if timing port provisioned AUTO	"SET-SYNCN: Set synchronization" (p. 3-3158)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, SSM not supported for this format	SSM not supported	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, SSM not supported for this format	SSM not supported for unframed formats	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, switch capacity cannot be decreased	No decrease of maximum switch capacity	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
IDNV - Input, Data Not Valid, TCA profile does not exist	profile name does not exist	"RTRV-TCA-ASGNMT: Retrieve TCA assignment" (p. 3-2979)
IDNV - Input, Data Not Valid, TCA profile does not exist	TCA profile exists	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-EPORT: Edit Ethernet port" (p. 3-329); "ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411); "ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-OC48T: Edit OC48 transparent port" (p. 3-648); "ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM16T: Edit STM16T port" (p. 3-1097); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111); "ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-T3: Edit DS3 port" (p. 3-1202); "ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card" (p. 3-1306); "ED-VC3: Edit VC3 tributary" (p. 3-1339); "ED-VCG: Edit virtual concatenation group" (p. 3-1354); "ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application" (p. 3-1405); "ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
IDNV - Input, Data Not Valid, TID of local node not allowed	Local node TID not allowed	"ED-FETID: Edit far end target identifier" (p. 3-405)
IDNV - Input, Data Not Valid, TNA Address already assigned to UNI port	TNA parameter disjointness between EDGE and UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, TNA Address not allowed for the Client Node ID	Network TNA Address Uniqueness per UNI client;	"ED-STM1E: Edit STM1E port" (p. 3-870); "ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, TNA already assigned to a VCG port	TNA parameters already assigned on VCAT layer;	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, TNA Type and TNA Address do not match	(ONNS specific): Check of TNA parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, TNA Type and TNA Address do not match	(ONNS specific): Check of VCAT layer TNA parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, TNA Type and TNA Address do not match	Check of TNA parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, Transparent Ports not allowed as line timing reference	No transparent port as line timing reference	"SET-SYNCN: Set synchronization" (p. 3-3158)
IDNV - Input, Data Not Valid, tributary already part of a cross-connection	(ONNS specific): loca, locz changes only allowed when tributary not involved in a cross-connection	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
IDNV - Input, Data Not Valid, uid must not be the own uid	uid must not be the own uid	"CANC-USER-SECU: Cancel user security" (p. 3-62)

Error code	Meaning (validation rule)	Related to command
IDNV - Input, Data Not Valid, uid not in NE data base	uid not in NE data base	"CANC-USER-SECU: Cancel user security" (p. 3-62)
IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT value	Check of UNI parameter	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT values	Check of UNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, UNI parameters must be different to their DEFAULT values	Check of UNI parameters;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, UNI port parameters must be different to their DEFAULT values	Check of UNI parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, Valid values for ENNI parameters required	Check of ENNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
IDNV - Input, Data Not Valid, Valid values for UNI parameters required	Check of UNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IDNV - Input, Data Not Valid, VCG already exists	VCG already exists	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)

Error code	Meaning (validation rule)	Related to command
`IDNV - Input, Data Not Valid, wrong old_pid`	Invalid old_pid	"ED-PID: Edit password identifier" (p. 3-798)
`IDNV - Input, Data Not Valid, zero values can only be set for all neighbor information consistently`	Setting of neighbor information to UNKNOWN	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`IDNV - Input, Data Not Valid. Attempting to delete an administrator login. Two administrator user logins are required at all times`	Attempt to delete administrator's uid	"DLT-USER-SECU: Delete user security" (p. 3-272)
`IDNV - UNI parameters must be different to their DEFAULT values`	Check of UNI parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IDRG - Input, Data Range Error`	Check aid and dyn consistency	"ENT-LONIM notification" (p. 3-1598); "RTRV-LONIM: Low-order non-intrusive monitor" (p. 3-2486); "RTRV-LOPTF: Retrieve low-order path termination function" (p. 3-2478)
`IDRG - Input, Data Range Error`	Input Parameter Range Incorrect (IDRG)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IDRG - Input, Data Range Error, osinode`	TARP data cache enabling only valid for node-1	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`IDRG - Input, Data Range Error, osinode`	TARP origination only valid for node-1	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`IDRG - Input, Data Range Error, osinode`	TARP responder only valid for node-1	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`IDRG - Input, Data Range Error, Value not supported for this modifier and istd`	Improper Value	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`IEAE - Input, Entity Already Exists`	Entity already exists (IEAE)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IEAE - Input, Entity Already Exists, cross-connection exists on test tributary`	Cross-connection exists on the test tributary	"CONN-TACC: Connect test access" (p. 3-94)

Error code	Meaning (validation rule)	Related to command
IEAE - Input, Entity Already Exists, cross-connection exists on test tributary	Not allowed for bridged cross-connections;	"CONN-TACC: Connect test access" (p. 3-94)
IEAE - Input, Entity Already Exists, duplicate profile name	Duplicate newname	"ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
IEAE - Input, Entity Already Exists, duplicate profile name	Duplicate pfname	"ENT-ASAP-PROF: Enter alarm severity assignment profile" (p. 3-1446)
IEAE - Input, Entity Already Exists, duplicate profile name	profile name already exists	"ED-TCA-PROF: Edit TCA profile" (p. 3-1214); "ENT-TCA-PROF: Enter TCA Profile" (p. 3-1894)
IEAE - Input, Entity Already Exists, osinode	OSI Node is already created	"ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
IEAE - Input, Entity Already Exists, test access connection exists	Test tributary already connected	"CONN-TACC: Connect test access" (p. 3-94)
IENE - IENE - Input, Entity Not Exists, Specified crossconnection does not exist	LO cross-connection with specified topology must exist	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IENE - Input Entity Not Exists, Specified crossconnection does not exist	Entity (cross-connection) must exist	"ED-NNCRS: Edit NN cross-connection" (p. 3-457)
IENE - Input Entity Not Exists, Specified cross-connection does not exist	Entity (cross-connection) must exist	"DLT-CRS: Delete cross-connection" (p. 3-145); "ED-CRS: Edit cross-connection" (p. 3-292)
IENE - Input Entity Not Exists, Specified Transparent DCC Cross Connection does not exist	Entity (transparent DCC cross-connection) must exist	"DLT-CRS-FECOM: Delete transparent DCC cross-connection" (p. 3-156)
IENE - Input Entity Not Exists, Specified tributary/tributaries do not exist	Entities (tributaries) must exist	"DLT-CRS: Delete cross-connection" (p. 3-145); "ED-CRS: Edit cross-connection" (p. 3-292); "ED-NNCRS: Edit NN cross-connection" (p. 3-457)
IENE - Input Entity Not Exists, Specified tributary/tributaries do not exist	Entities must exist for HO cross-connections	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IENE - Input, Entity Not Exists	Check CCB against ptype and eif	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-T3: Edit DS3 port" (p. 3-1202); "RTRV-EC1: Retrieve EC1 port" (p. 3-2332)
IENE - Input, Entity Not Exists	Check LONIM::modifier against PTF::istd	"ENT-LONIM notification" (p. 3-1598)

Error code	Meaning (validation rule)	Related to command
IENE - Input, Entity Not Exists	Check LONIM::modifier against PTF::ptfstd	"RTRV-LONIM: Low-order non-intrusive monitor" (p. 3-2486)
IENE - Input, Entity Not Exists	Check modifier against circuit pack type	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416); "OPR-VCGTRIB: Operate virtual concatenation group tributary" (p. 3-2082); "RTRV-VCGTRIB: Retrieve virtual concatenation group tributary " (p. 3-3109)
IENE - Input, Entity Not Exists	Command only allowed for ES64U	"RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group" (p. 3-2819)
IENE - Input, Entity Not Exists	Command only allowed for VCG on ES64U circuit pack	"DLT-VCG – Enter virtual concatenation group" (p. 3-277)
IENE - Input, Entity Not Exists	Command only allowed for VCG on GEn or ES64U circuit packs	"RTRV-VCG: Retrieve virtual concatenation group" (p. 3-3078)
IENE - Input, Entity Not Exists	command only allowed for VCG on GEn or ES64U circuit packs;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IENE - Input, Entity Not Exists	Command only allowed for VCG on Gigabit Ethernet Packs	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
IENE - Input, Entity Not Exists	Command only allowed for VCG on Transparent Packs	"ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application" (p. 3-1405); "RTRV-VCG-TRANS: Retrieve virtual concatenation group transparent" (p. 3-3102)
IENE - Input, Entity Not Exists	Directory not accessible	"CPY-MEM: Copy memory" (p. 3-114)
IENE - Input, Entity Not Exists	Invalid Entity (IENE)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IENE - Input, Entity Not Exists	level 1 Routing controller does not exist	"CANC-L1ROUTING" (p. 3-54)
IENE - Input, Entity Not Exists	Protection Group must exist	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "OPR-PROTNSW: Operate protection switch" (p. 3-2057); "RLS-PROTNSW: Release protection switch" (p. 3-2203)
IENE - Input, Entity Not Exists	Reject Request for CTL slots	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547)
IENE - Input, Entity Not Exists, aid not part of substructure	Check aid against PTF substructure	"ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)

Error code	Meaning (validation rule)	Related to command
IENE - Input, Entity Not Exists, Command not allowed for a protecting ES64U pack	Command not allowed for ES64U protecting pack	"RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group" (p. 3-2819)
IENE - Input, Entity Not Exists, Command not allowed for a protecting ES64U pack	Command not allowed for ES64U protecting pack	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)
IENE - Input, Entity Not Exists, Command not allowed for an inactive ES64U pack	Command not allowed for ES64U with switch capacity 0	"RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group" (p. 3-2819)
IENE - Input, Entity Not Exists, Command not allowed for an inactive ES64U pack	Command not allowed for ES64U with switch capacity 0	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)
IENE - Input, Entity Not Exists, Command only allowed for a ES64U pack	Command only allowed for VCG on ES64U circuit pack	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)
IENE - Input, Entity Not Exists, constituent member does not exist	Boundary consistency for constituent members	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
IENE - Input, Entity Not Exists, Not for this interface standard	Entities must fit to the interface standard	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IENE - Input, Entity Not Exists, Specified association does not exist	Association present (when dis-associating)	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
IENE - Input, Entity Not Exists, Specified crossconnection does not exist	Conversion cross-connection with specified topology must exist	"DLT-CRS: Delete cross-connection" (p. 3-145)
IENE - Input, Entity Not Exists, Specified crossconnection does not exist	Entity (cross-connection) must exist	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IENE - Input, Entity Not Exists, Specified cross-connection does not exist	LO cross-connection with specified topology must exist	"DLT-CRS: Delete cross-connection" (p. 3-145)
IENE - Input, Entity Not Exists, Specified low-order tributary/ tributaries do not exist/cannot be created (possibly connected at high-order level)	LO tributaries must be creatable for conversion cross-connections	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IENE - Input, Entity Not Exists, Specified OUC does not exist	Entity (OUC) must exist	"DLT-OUC: Delete ONNS to UPSR/SNCP constructs" (p. 3-217)
IENE - Input, Entity Not Exists, Specified port does not exist	Port exists (when associating)	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	Entities (tributaries) must exist	"RLS-LPBK: Release loopback (cross-connection)" (p. 3-2195)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	Entities must exist	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)

Error code	Meaning (validation rule)	Related to command
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	HO tributaries must exist	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	HO tributaries must exist for LO TACC sessions	"CONN-TACC: Connect test access" (p. 3-94)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	HO tributary must exist for conversion cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	LO tributary must exist/be creatable	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	No LO tributary at HO side of conversion cross-connection	"ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	Tributaries must exist	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
IENE - Input, Entity Not Exists, Specified tributary/tributaries do not exist	Tributaries must exist for HO TACC sessions	"CONN-TACC: Connect test access" (p. 3-94)
IENE - Input, Entity Not Exists, Substructure cannot be adapted	Entities must be adaptable for LO cross-connections	"ENT-CRS: Enter cross-connection" (p. 3-1464)
IENE - Input, Entity Not Exists, Substructure cannot be adapted	Entities must be adaptable for LO loopbacks	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
IENE - Input, Entity Not Exists, Substructure cannot be adapted	Entity must be adaptable for LO cross-connections	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
IIAC - Input, Invalid ACcess identifier, incorrect syntax	consistent aid and protype	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
IIAC - Input, Invalid ACcess identifier, incorrect syntax	Incorrect AID Syntax (IIAC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IIAC - Input, Invalid ACcess identifier, unknown AID	Unknown AID (IIAC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IIAC - Input, Invalid ACcess identifier,incorrect syntax	consistent aid and protype	"RLS-PROTNSW: Release protection switch" (p. 3-2203)
IIAC - Input, Invalid ACcess identifier,incorrect syntax	consistent aid and rid	"RLS-PROTNSW: Release protection switch" (p. 3-2203)
IICT - Input, Invalid Correlation Tag (CTAG), incorrect syntax	Invalid CTAG (IICT)	"RTRV-HDR: Retrieve header" (p. 3-2417)
IISP - Input, Invalid Syntax or Punctuation	Syntax or Punctuation Error (IISP)	"RTRV-HDR: Retrieve header" (p. 3-2417)

Error code	Meaning (validation rule)	Related to command
`IISP - Input, Invalid Syntax or Punctuation, command too long`	Command too long	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IITA - Input, Invalid TArget identifier, incorrect syntax`	Incorrect TID (IITA)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IITA - Input, Invalid TID, Communications Failure`	Not reachable TID (IITA)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IITA - Input, Invalid TID, Unknown TID`	Unknown TID (IITA)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`INUP - Input, Non-null Unimplemented Parameter`	Check if parameters are supported for the modifier	"ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)
`INUP - Input, Non-null Unimplemented Parameter`	Less than 2 nodes specified	"ED-MAP-RING: Edit map ring" (p. 3-436)
`INUP - Input, Non-null Unimplemented Parameter`	Parameter Input Failure (INUP)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`INUP - Input, Non-null Unimplemented Parameter, secondary test tributary not allowed`	Secondary test tributary not needed for MONE test modes	"CONN-TACC: Connect test access" (p. 3-94)
`INVS - Status, Not in Valid State, TARP propagation can not be disabled`	Can't disable TARP propagation while LAN TARP storm suppression is enabled	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`IPMS - Input, Parameter MiSsing, abstractNodeID is missing`	Abstract Node ID missing	"ACT-L1ROUTING" (p. 3-15)
`IPMS - Input, Parameter MiSsing, <parameter_name> is missing`	Parameter Missing (IPMS)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`IPMS -Input, Parameter MiSsing, <parameter name> is missing`	Path-protected crossconnection: missing parameters	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`IPMS -Input, Parameter Missing, <parameter name> missing`	path-protected cross-connection: parameters missing	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`IPNC - Input Parameter Not Consistent, Rate inconsistent with cross-connection rate`	Only those parameter present which are needed	"ED-CRS: Edit cross-connection" (p. 3-292)
`IPNC - Input Parameter Not Consistent, Rate inconsistent with cross-connection rate`	Specified rate consistent with the cross-connecton rate	"ED-CRS: Edit cross-connection" (p. 3-292); "ED-NNCRS: Edit NN cross-connection" (p. 3-457)
`IPNC - Input Parameter Not Consistent, TID not compliant with cross-connection configuration`	TID validation	"ED-CRS: Edit cross-connection" (p. 3-292)
`IPNC - Input, Parameter Not Consistent`	Invalid Command Qualifier (IPNC)	"RTRV-HDR: Retrieve header" (p. 3-2417)

Error code	Meaning (validation rule)	Related to command
`IPNC - Input, Parameter Not Consistent, 64 byte path trace not allowed`	Check modifier and ptrcrfmt consistency	"ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411); "ED-LOPTF: Edit lower-order path termination function" (p. 3-427)
`IPNC - Input, Parameter Not Consistent, Rate inconsistent with crossconnection rate`	Inconsistent tributary rate	"CHG-ACCMD: Change access mode" (p. 3-65); "CONN-TACC: Connect test access" (p. 3-94); "DISC-TACC: Disconnect test access" (p. 3-129)
`IPNC - Input, Parameter Not Consistent, SES threshold out of range for this modifier`	Check modifier and pmlpneses/pmlpnefeses consistency	"ED-LONIM: Edit lower-order non-intrusive monitor" (p. 3-411)
`PICC - Privilege, Illegal Command Code`	Command requires higher privilege (PICC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`PICC - Privilege, Illegal Command Code, invalid user class`	user class	"RTRV-AO: Retrieve autonomous output" (p. 3-2241)
`PLNA - Privilege, Login Not Active`	Session is not initiated (PLNA)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SABT - Status, Aborted`	Command Aborted (SABT)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SAOP - Status Already Operated, loopback already exists`	Loopback must not already exist	"OPR-LPBK: Operate facility loopback" (p. 3-2044); "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SARB - Status, All Resources Busy`	APPLY already in progress	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SARB - Status, All Resources Busy`	Maximum logins exceeded	"ACT-USER: Activate user" (p. 3-24)
`SARB - Status, All Resources Busy`	System Busy with SW Generic Installation (SARB)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SARB - Status, All Resources Busy`	Temporary exhaustion of resources	"COPY-RFILE: Copy remote file" (p. 3-106); "CPY-MEM: Copy memory" (p. 3-114)
`SARB - Status, All Resources Busy, LOXC capacity exhausted`	No Spare capacity for HO cross-connection legs on LOXC	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SARB - Status, All Resources Busy, LOXC capacity exhausted`	Spare capacity for HO cross-connection legs on LOXC	"CONN-TACC: Connect test access" (p. 3-94)
`SARB - Status, All Resources Busy, system limit exceeded`	System Limit exceeded (SARB)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SARB - Status, All Resources Busy, system limit exceeded`	Temporary system exhaustion	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)

Error code	Meaning (validation rule)	Related to command
SCNF - Status, Command Not Found, command not recognized or command parameters not valid	Command not recognized (SCNF)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SDNC - Status, Data Not Consistent	Additional directory/URL specified	"COPY-RFILE: Copy remote file" (p. 3-106); "CPY-MEM: Copy memory" (p. 3-114)
SDNC - Status, Data Not Consistent	Additional directory/URL specified for backup	"COPY-RFILE: Copy remote file" (p. 3-106)
SDNC - Status, Data Not Consistent	Additional directory/URL specified for database download	"COPY-RFILE: Copy remote file" (p. 3-106)
SDNC - Status, Data Not Consistent	Additional directory/URL specified for restoration	"COPY-RFILE: Copy remote file" (p. 3-106)
SDNC - Status, Data Not Consistent	File inappropriate	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SDNC - Status, Data Not Consistent	Source product type does not match network product type	"CPY-MEM: Copy memory" (p. 3-114)
SDNC - Status, Data Not Consistent, asgncond not empty	probable cause assigned to MDO port	"OPR-EXT-CONT: Operate external control" (p. 3-2032); "RLS-EXT-CONT: Release external control" (p. 3-2190)
SDNC - Status, Data Not Consistent, invalid combination of osinode parameter with dccstat, scn_ dccstat	(SCN specific): Check combination of osinode parameter with scn_dccstat	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SDNC - Status, Data Not Consistent, invalid combination of osinode parameter with dccstat, scn_ dccstat, uni_dccstat, enni_dccstat	(UNI/E-NNI specific): Check combination of osinode parameter with uni_dccstat / enni_dccstat	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SDNC - Status, Data Not Consistent, invalid file or directory	Invalid file or directory name	"CPY-MEM: Copy memory" (p. 3-114)
SDNC - Status, Data Not Consistent, invalid file or directory	Invalid File or Directory Name (SDNC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SDNC - Status, Data Not Consistent, invalid instance of entity	Invalid Instance (SDNC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SDNC - Status, Data Not Consistent, invalid instance of profile	pfname does not exist	"DLT-ASAP-PROF: Delete alarm severity assignment profile" (p. 3-134); "ED-ASAP-PROF: Edit alarm severity assignment profile" (p. 3-284)
SDNC - Status, Data Not Consistent, Not allowed on ES64U pack	;	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)

Error code	Meaning (validation rule)	Related to command
`SDNC - Status, Data Not Consistent, Not allowed on ES64U pack`	Path trace not supported on ES64U;	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
`SDNC - Status, Data Not Consistent, Not allowed on ES64U pack`	Trail signal label mismatch monitor not supported on ES64U;	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
`SDNC - Status, Data Not Consistent, Not allowed on GE10PL1 pack`	64 byte path trace setting on GE10PL1 only for first VCG tributary of a VCG	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
`SDNC - Status, Data Not Consistent, Not allowed on GE10PL1 pack`	Path trace monitoring on GE10PL1 only for first VCG tributary of a VCG	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
`SDNC - Status, Data Not Consistent, other port already assigned`	No timing port on protection pack of an MS protected timing port	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, other port already assigned`	One timing port for a circuit pack	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, pgid is already used by another protection group of the same type`	unique pgid	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SDNC - Status, Data Not Consistent, port already assigned`	Port is already assigned to another timing reference	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, port is out-of-service`	Check for service condition	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SDNC - Status, Data Not Consistent, port is out-of-service`	No out-of-service state for timing port	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, profile in use`	ASAP in use	"DLT-ASAP-PROF: Delete alarm severity assignment profile" (p. 3-134)
`SDNC - Status, Data Not Consistent, quality evaluation disabled`	Command rejection if quality evaluation disabled	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, reference already assigned`	Timing reference already assigned	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SDNC - Status, Data Not Consistent, reference in protection group`	Timing reference from a protection port in a protection group only allowed in asymmetric spans of a ring	"SET-SYNCN: Set synchronization" (p. 3-3158)
`SIPC - E091::Status, Invalid command for protection class MBYNWRK or MBYNPRT`	protectionClass is not allowed	"ED-PATH-PROT: Modify ONNS connection parameters or protection type" (p. 3-753)
`SLEM - Status, List Exceeds Maximum UIDs`	Maximum number of uids reached	"ENT-USER-SECU: Enter user security" (p. 3-1970)

Error code	Meaning (validation rule)	Related to command
SNVS - Not in Valid State, DCC needed for BLSR/MS-SPRING	DCC for 2 fiber BLSR/MS-SPRING	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Not in Valid State, DCC needed for BLSR/MS-SPRING	Protection port for 4 fiber BLSR/MS-SPRING	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Not in Valid State, DCC needed for BLSR/MS-SPRING	Worker port for 4 fiber BLSR/MS-SPRING	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Not in Valid State, DCC protection available	Check for DCC Protection	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Not in Valid State, part of 1+1 MSP protection	DCC Protection	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections are still existing	Setting of portClass to ENNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	Setting of portClass to ENNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - SNVS - Status, Not in Valid State, LOXC switch capacity invalid	No lower order TransMUX cross-connections when LO switch capacity is lower than 40G	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State	Check for enabled DCC channels	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Status, Not in Valid State	Check of the maintenance condition	"ACT-L1ROUTING" (p. 3-15); "CANC-L1ROUTING" (p. 3-54); "ED-RSVP: Modify RSVP parameters" (p. 3-857)
SNVS - Status, Not in Valid State	Command denied based on Mode, State or Status (SNVS)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SNVS - Status, Not in Valid State	Deemed source file	"CPY-MEM: Copy memory" (p. 3-114)
SNVS - Status, Not in Valid State	ip_lanstat must be ENABLE for uninode and ennimode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State	isis_stat must be DISABLE for uninode and ennimode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State`	lanstat must be ENABLE for uninode and ennimode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State`	No SW installation during DB backup, DB download or DB restore	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SNVS - Status, Not in Valid State`	osi_lanstat must be ENABLE for enninode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State`	osi_lanstat must be ENABLE for uninode	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State`	Restrictions for conversion to LO path protection	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`SNVS - Status, Not in Valid State`	Restrictions for LO path protection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State`	Second APPLY on same software	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SNVS - Status, Not in Valid State, Auto-Encapsulation must be switched on`	Auto-encapsulation must be enabled ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, DCN IP Address is not configured`	dcn_ipad must be configured ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, Pack is protecting pack`	ESdcn_ipad setting not for card in protection functionality;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, ES64U switch capacity invalid`	ES64U switch capacity and unit roles	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	(ONNS specific): Change the eportClass to UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	Change the portClass to EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	Change the portClass to INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, a cross connect loopback exists`	No XC loopbacks allowed	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
`SNVS - Status, Not in Valid State, a cross-connect loopback exists`	no XC loopbacks for interface type change	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, a loopback is existing`	no loopbacks	"DLT-VCG – Enter virtual concatenation group" (p. 3-277)
`SNVS - Status, Not in Valid State, a loopback is existing`	no loopbacks for interface standard change	"ED-VCG: Edit virtual concatenation group" (p. 3-1354); "ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application" (p. 3-1405)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, a loopback is existing	no loopbacks for istd change	“ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111)
SNVS - Status, Not in Valid State, a loopback is existing	no loopbacks for istd, trbmd change	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585)
SNVS - Status, Not in Valid State, a loopback is existing	Not allowed if a member is looped	“ED-VCG: Edit virtual concatenation group” (p. 3-1354)
SNVS - Status, Not in Valid State, a path protection group is existing	no path protection for trbmd change to ADAPTIVE	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585)
SNVS - Status, Not in Valid State, a protection group is existing	no protection for istd change	“ED-STM1: Edit STM1 port” (p. 3-915); “ED-STM16: Edit STM16 port” (p. 3-1035); “ED-STM4: Edit STM4 port” (p. 3-975); “ED-STM64: Edit STM64 port” (p. 3-1111)
SNVS - Status, Not in Valid State, a protection group is existing	no protection for istd, trbmd change	“ED-OC12: Edit OC12 port” (p. 3-523); “ED-OC192: Edit OC192 port” (p. 3-663); “ED-OC3: Edit OC3 port” (p. 3-461); “ED-OC48: Edit OC48 port” (p. 3-585)
SNVS - Status, Not in Valid State, a protection group is existing	No protection groups allowed	“DLT-EQPT (slot): Delete equipment (slot)” (p. 3-177)
SNVS - Status, Not in Valid State, a test access session exists	(ONNS specific): Change the eportClass to UNI	“ED-VCG: Edit virtual concatenation group” (p. 3-1354)
SNVS - Status, Not in Valid State, a test access session exists	(ONNS specific): Setting of eportClass to EDGE only possible, if no test access session actve	“ED-VCG: Edit virtual concatenation group” (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, a test access session exists	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, a test access session exists	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, a test access session exists	No interface standard change on GE10PL1 pack with existing test access session	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SNVS - Status, Not in Valid State, a test access session exists	no test access session for interface type change	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-T3: Edit DS3 port" (p. 3-1202)
SNVS - Status, Not in Valid State, a test access session exists	no test access session for istd change	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, a test access session exists	no test access session for istd, trbmd change	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, a test access session exists	Not allowed if involved in test access	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, a test access session exists	Setting of portClass to INNI only possible, if no test access session active	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, a test access session exists	Setting of portClass to INNI or EDGE only possible, if no test access session active	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Active Repeater Mode for this Port does not allow dftvlan to be modified	Default VLAN ID not allowed in repeater mode	"ED-EPORT: Edit Ethernet port" (p. 3-329)
SNVS - Status, Not in Valid State, Active Repeater Mode for this Port does not allow vvidls to be modified	VLAN filtering database not allowed in repeater mode	"ED-EPORT: Edit Ethernet port" (p. 3-329)
SNVS - Status, Not in Valid State, Assigned Node not allowed for IP	Assignmet of OsiNode without IP support	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, Assigned Node not allowed for IP	only osinode1, scnnode, uninode and enninode supports IP	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, association already exists	No association present (when associating)	"SET-OW: Set orderwire and user bytes" (p. 3-3147)
SNVS - Status, Not in Valid State, at least one port of the protection pack is used as timing reference	Not used as timing reference	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, at least one port used as timing reference	Not used as timing reference	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Auto Encapsulation not enabled	Auto Encapsulation must be enabled	"ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
SNVS - Status, Not in Valid State, Auto Encapsulation still in use	Check if Auto Encapsulation is still in use by ES64U pack;	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
SNVS - Status, Not in Valid State, Backplane serial number mismatch	Serial number in the database not equal to system backplane serial number	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SNVS - Status, Not in Valid State, Broadcast with N greater 2 exists	Change the portClass to EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, can not change portClass must not be UNI	Some Parameter Values not settable for ports of portClass UNI	"ED-STM1E: Edit STM1E port" (p. 3-870)
SNVS - Status, Not in Valid State, can not change servcond for this port class	Check of parameters for ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, can not change servcond for this port class	Check of parameters for UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, can not disable servcond for INNI ports`	servcond must be enabled when portClass = INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, can not disable servcond for protection ports`	servcond must be enabled for protection ports	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Cannot change servcond for this port class`	Change the portClass away from UNI	"ED-STM1E: Edit STM1E port" (p. 3-870)
`SNVS - Status, Not in Valid State, cannot disable IP on all LAN ports while T-TD or FTTD is enabled`	IP disabling while T-TD or FTTD is enabled	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group`	ES64U pack already used in other equipment protection group	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group`	TransMUX pack already used in other equipment protection group	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group`	Check whether TransMUX circuit pack is member of an equipment protection group	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group`	Check whether ES64U circuit pack is member of an equipment protection group ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group`	Check whether LOXC circuit pack is member of an equipment protection group	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, circuit pack is member of an equipment protection group	EPn pack already used in other equipment protection group	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, circuit packs not provisioned	Circuit pack not provisioned	"CONN-TACC: Connect test access" (p. 3-94)
SNVS - Status, Not in Valid State, clear request impossible	Clear wait to restore timer not applicable	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
SNVS - Status, Not in Valid State, clock of destination XC is in warming-up state, please wait some minutes	Clock not in warming-up status (Switching Protection Group)	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, Command refers to protection tributaries of a BLSR/ MS-SPRing protection	2 Fiber BLSR/MS-SPRing protection access (Extra Traffic)	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Command refers to protection tributaries of a BLSR/ MS-SPRing protection	4 Fiber BLSR/MS-SPRing protection access (Extra Traffic)	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Command refers to protection tributaries of a BLSR/ MS-SPRing protection	BLSR/MS-SPRing protection access (Extra Traffic)	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, Command refers to protection tributaries of a BLSR/ MS-SPRing protection	loca/locz information only retrievable for EXTRA TRAFFIC	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, command supported for MS-SPRing TOPEX only	RTRV-SPANSWS only for TOPEX	"RTRV-SPANSWS: Retrieve SPAN switch state" (p. 3-2830)
SNVS - Status, Not in Valid State, Conflicting OUC already exists	No conflicting OUC must already exist between working and protection tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, connection to OW port exists	Not used for OW	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, connection to OW port of the protection pack exists	Not used for OW	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, conversion cross-connections are still existing	Command not allowed while conversion cross-connections exist	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, conversion cross-connections are still existing`	newinaid must not be involved in conversion cross-connections	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`SNVS - Status, Not in Valid State, conversion cross-connections are still existing`	no conversion cross-connections for interface type change	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, conversion cross-connections are still existing`	No LO cross-connections while conversion cross-connections exist	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, conversion cross-connections are still existing`	No LO loopback while conversion cross-connections exist	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, conversion cross-connections are still existing`	Substructure change not allowed while conversion cross-connections exist	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)
`SNVS - Status, Not in Valid State, conversion cross-connections not allowed with LOXC/1`	No conversion cross-connections with LOXC/1	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, cross connect loopback exists`	Change the portClass to ENNI	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Cross Connection of xconcls NN exists`	(ONNS specific): TRADITIONAL and NN cross-connections not on the same tributary	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, Cross-connection involved in Test Access Session`	Cross-connection not involved in Test Access Session	"ED-NNCRS: Edit NN cross-connection" (p. 3-457)
`SNVS - Status, Not in Valid State, Cross-connection must not be of xconcls NN`	Cross-connection must not be of xconcls NN	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "DLT-CRS: Delete cross-connection" (p. 3-145); "ED-CRS: Edit cross-connection" (p. 3-292)
`SNVS - Status, Not in Valid State, Cross-connection not of class NN`	Cross-connection must have xconcls = NN	"ED-NNCRS: Edit NN cross-connection" (p. 3-457)
`SNVS - Status, Not in Valid State, cross-connections are still existing`	crossconnection - interface standard change	"ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application" (p. 3-1405)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, cross-connections are still existing`	No cross-connections allowed	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
`SNVS - Status, Not in Valid State, cross-connections are still existing`	Not allowed if a member is cross-connected	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Cross-connections are still existing`	crossconnection - interface standard change	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, cross-connections or substructure still exist`	No HO/LO cross-connection or LO substructure at protection tribs	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, cross-connections or substructure still exist`	No HO/LO cross-connection or LO substructure at tributaries	"DLT-PROTN-GRP: Delete protection group" (p. 3-233); "ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, cross-connections still exist`	no cross-connections for interface type change	"ED-EC1: Edit EC1 port" (p. 3-317)
`SNVS - Status, Not in Valid State, cross-connections still existing`	no cross-connections for interface type change	"ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	(ONNS specific): Change the eportClass to UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Cross-connections still existing`	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	Check whether ES64U circuit pack is in use;	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	Modify of ENNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	Modify of UNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111); "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Cross-connections still existing`	no crossconnection	"DLT-VCG – Enter virtual concatenation group" (p. 3-277)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Cross-connections still existing	no cross-connections for istd change	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Cross-connections still existing	no cross-connections for istd, trbmd change	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, Cross-connections still existing	No modification of trbmd at ports of portClass ENNI with existing cross-connections	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, Cross-connections still existing	No modification of trbmd to ADAPTIVE at ports of portClass EDGE with existing ONNS cross-connections	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, Cross-connections still existing	No modification of trbmd to FIXED at ports of portClass EDGE with existing ONNS cross-connections	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, Cross-connections still existing	Setting of portClass to INNI only possible, if no cross-connection existing	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Database already installed	Second APPLY on previous database	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SNVS - Status, Not in Valid State, database corrupt or not available	APPLY applied on corrupted database	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, database not compatible`	No DB restore with incompatible database	"COPY-RFILE: Copy remote file" (p. 3-106); "CPY-MEM: Copy memory" (p. 3-114)
`SNVS - Status, Not in Valid State, DCC channel already enabled`	DCC channels enabled	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
`SNVS - Status, Not in Valid State, DCC channel can not be assigned to more than one of MCN, SCN, UNI and E-NNI at the same time`	Independent use of DCC channel for MCN, SCN, UNI, or E-NNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SNVS - Status, Not in Valid State, DCC channel occupied by MSP/APS DCC protection`	DCC protection	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
`SNVS - Status, Not in Valid State, DCC channel used in transparent DCC cross-connection`	Transparent DCC cross-connection	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SNVS - Status, Not in Valid State, DCC channel used in transparent DCC cross-connection`	Transparent DCC for DCC Line/MS protection mode	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, DCC channel used in transparent DCC cross-connection`	Transparent DCC for DCC Sect/RS protection mode	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, DCC channels are active for this port`	servcond must be enabled when DCC enabled	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, DCC must not be assigned to ENNINode`	Change the portClass away from ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, DCC must not be assigned to UNINode`	Change the portClass away from UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, DCC not in use for ENNI`	DCC port not used for E-NNI	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528)
`SNVS - Status, Not in Valid State, DCC on SCN may not be enabled`	Changing of portClass from INNI to TRADITIONAL not possible, if DCC for SCN is enabled on the port	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, dcc protection side carries traffic`	dcc protection: protection side is active	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, DCC/LAN channels assigned`	Reject deletion when node is assigned to LAN or DCC	"DLT-ULSDCC-L3: Delete a node ID other than node-1" (p. 3-265)
`SNVS - Status, Not in Valid State, default router LAN port not configured`	check for correct configured Default Router LAN Port	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
`SNVS - Status, Not in Valid State, Different interface standards for input tributaries`	path-protected cross-connection has inputs from ports with same interface standard	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`SNVS - Status, Not in Valid State, Different interface standards for input tributaries`	path-protected cross-connection has inputs from ports with same interface standards	"ENT-CRS: Enter cross-connection" (p. 3-1464)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Different interface standards for working and protection tributaries`	Tributaries must belong to ports of equal interface standards	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, DS3 path PM not allowed with Unframed Clear-Channel`	DS3 path PM not allowed with unframed clear-channel	"ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, DS3AIS or IDLE not allowed with Unframed Clear-Channel`	DS3AIS or IDLE not allowed with unframed clear-channel	"ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, Duplex Control Recovery in progress`	No DB activation during Duplex Control Recovery	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SNVS - Status, Not in Valid State, EDGE port carrying EDGE-prt must not be in LOCKOUT`	Affected EDGE port the EDGE-prt tributary is located on must not be in LOCKOUT state	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, EDGE ports must not be in line or equipment protection schemes`	Affected EDGE ports must not be in a line or equipment protection scheme	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, ENNI node must exist`	Setting of TNA information only if ENNI Node exists	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, ENNI Node not created`	(UNI/E-NNI specific): ENNI node must be created	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SNVS - Status, Not in Valid State, ENNI Node not created`	ENNI node must be created	"ED-RSVP: Modify RSVP parameters" (p. 3-857)
`SNVS - Status, Not in Valid State, E-NNI Node not created`	(UNI/E-NNI specific): E-NNI Node must be created	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SNVS - Status, Not in Valid State, E-NNI Node not created`	ENNI node must be created	"RTRV-RSVP: Retrieve RSVP parameters" (p. 3-2823)
`SNVS - Status, Not in Valid State, E-NNI Node not created`	E-NNI Node must be created	"RTRV-ENNIMANROUTE: Retrieve all manual routes of the E-NNI node" (p. 3-2348)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, ES64U destination not usable	No forced switch to unusable ES64U pack (ES64U Protection Group);	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, extra traffic cross-connections are still existing	No cross-connections allowed when leaving TOPEX	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, Facility Loopback is active for this port	servcond must be disabled for facility loopback	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-EPORT: Edit Ethernet port" (p. 3-329); "ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-OC48T: Edit OC48 transparent port" (p. 3-648); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM16T: Edit STM16T port" (p. 3-1097); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111); "ED-T3: Edit DS3 port" (p. 3-1202)
SNVS - Status, Not in Valid State, Facility loopback with different type exists at neighbor port	Inconsistent facility loopback types on port pairs	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
SNVS - Status, Not in Valid State, facility loopbacks still exist	No facility loopbacks allowed	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, facility loopbacks still exist	No facility loopbacks for interface type change	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-T3: Edit DS3 port" (p. 3-1202)
SNVS - Status, Not in Valid State, facility loopbacks still exist at the protection pack	No facility loopbacks allowed	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, given TNA already exists	TNA parameters on VCAT layer already assigned on SDH/SONET layer;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, given TNA already exists	TNA parameters on VCAT layer already assigned to other UNI or EDGE port	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, given TNA already exists	TNA parameters on VCAT layer already assinged to other VCG port	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, given TNA already exists	Uniqueness of TNA parameters on VCAT layer per UNI VCG	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, HO cross-connection exists`	LO cross-connections not at HO tributaries used in HO cross-connections	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, HO cross-connection exists`	LO Substructure creation not allowed on already cross- connected tributaries	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
`SNVS - Status, Not in Valid State, HO cross-connection exists`	LO TACC sessions not at HO tributaries used in HO cross-connection	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, HO or LO cross-connections still exist`	Check whether TransMUX circuit pack is in use	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, HO or LO cross-connections still exist`	Check whether TransMUX pack is in use	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
`SNVS - Status, Not in Valid State, HO tributary already substructured`	LO Substructure creation not allowed on already substructured tributaries	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
`SNVS - Status, Not in Valid State, illegal local node identifier`	Illegal local node ID	"ED-MAP-RING: Edit map ring" (p. 3-436)
`SNVS - Status, Not in Valid State, implicit channel re-assignment`	Implicit status change of dccstat, scn_dccstat, uni_dccstat, or enni_dccstat	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
`SNVS - Status, Not in Valid State, incompatibility istddflt and trbmddflt`	compatibility istddflt and trbmddflt	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
`SNVS - Status, Not in Valid State, incomplete IP configuration`	check for correct IP configuration	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
`SNVS - Status, Not in Valid State, incomplete IP configuration`	Check for correct IP configuration	"ED-NE: Edit NE" (p. 3-442); "ENT-IP-MAP: Enter TCP/IP map" (p. 3-1594)
`SNVS - Status, Not in Valid State, incomplete IP configuration`	Check for IP configuration of LAN1;	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
`SNVS - Status, Not in Valid State, incomplete IP configuration`	Check IP configuration	"ACT-L1ROUTING" (p. 3-15)
`SNVS - Status, Not in Valid State, incomplete IP configuration`	Compatibility xcsetup and IP configuration for SCN	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
`SNVS - Status, Not in Valid State, Inconsistent Input Tributary Mode`	Inconsistent Tributary Mode of Ingress Port	"ENT-CONN: Establish network connections with TNA address parameters" (p. 3-1455)
`SNVS - Status, Not in Valid State, Input tributary shared with other path protection group`	SNC Type change on higher order path protection groups sharing input tributaries	"ED-PROTN-GRP: Edit protection group" (p. 3-803)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Interfering cross-connection existing on protection tributary`	No interfering cross-connection or OUC must exist on protection tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, invalid IP configuration`	Check IP configuration	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State, invalid test access mode`	Invalid test access mode change request	"CHG-ACCMD: Change access mode" (p. 3-65)
`SNVS - Status, Not in Valid State, invalid test access mode`	Underlying cross connection belongs to BLSR/MS-SPRing	"CHG-ACCMD: Change access mode" (p. 3-65)
`SNVS - Status, Not in Valid State, invalid test access mode`	Underlying cross connection is not two-way	"CHG-ACCMD: Change access mode" (p. 3-65)
`SNVS - Status, Not in Valid State, Invalid topology`	Valid topology	"DLT-CRS: Delete cross-connection" (p. 3-145)
`SNVS - Status, Not in Valid State, IP communication not enabled for LAN port`	IP communication must be enabled for LAN port	"ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)
`SNVS - Status, Not in Valid State, J0 parameters can not be set for INNI ports`	Some Parameters not settable for ports of portClass = INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, l3radesnode not in created stat`	l3radesnode must be in the created state	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
`SNVS - Status, Not in Valid State, l3radesnode not in created state`	l3radesnode must be in the created state	"ENT-ULSDCC-L3: Create a further OSI node" (p. 3-1951)
`SNVS - Status, Not in Valid State, LAN is in use for Default Router`	IP disabling while used as IP Default Router LAN	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State, LAN port not in use for ENNI`	LAN port not used for E-NNI	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, LAN port not in use for UNI`	LAN port not used for UNI	"DLT-MANROUTE-LAN: Delete manual route at LAN port" (p. 3-203); "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610); "RTRV-MANROUTE-LAN: Retrieve manual route at LAN port" (p. 3-2516)
`SNVS - Status, Not in Valid State, LAN1 can not be disabled`	Can't disable LAN1	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State, LO cross-connections or substructure still exist`	No conversion-cross-connections while LO cross-connections or LO substructure exist	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, LO cross-connections or substructure still exist`	No HO loopback while LO cross-connection, substructure or conversion cross-connection exists	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	Check whether LOXC circuit pack is in use	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	Check whether LOXC pack is in use	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	No HO cross-connections while LO cross-connections or LO substructure exist	"ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	No HO cross-connections while LO cross-connections, substructure or conversion cross-connections exist	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	No HO loopback while conversion cross-connection exists	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	No HO TACC sessions while LO cross-connection, substructure or conversion cross-connection exists	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist`	No HO/LO cross-connections, conversion cross-connections or substructure allowed	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	No lower order cross-connections, conversion cross-connections, loopbacks or substructure allowed	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	No lower order cross-connections, conversion cross-connections, loopbacks or substructure for istd change	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	No lower order cross-connections, conversion cross-connections, loopbacks or substructure for istd, trbmd change	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	Setting of portClass to INNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	Setting of portClass to UNI only possible, if no LO cross-connection, substructure or conversion cross-connections exist	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, LO cross-connections, substructure or conversion cross-connections still exist	Setting of portClass to UNI only possible, if no LO cross-connection, substructure, or conversion cross-connections exist	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, loopback does not exist	Loopback must exist	"RLS-LPBK: Release loopback (facility)" (p. 3-2198); "RLS-LPBK: Release loopback (cross-connection)" (p. 3-2195)
SNVS - Status, Not in Valid State, lower order cross-connect loopbacks are still existing	Substructure deletion not allowed with LO cross-connect loopbacks existing	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, lower order cross-connections are still existing	Substructure change not allowed with LO cross-connections at input tributaries	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)
SNVS - Status, Not in Valid State, lower order cross-connections are still existing	Substructure change not allowed with LO cross-connections at output tributaries	"ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830)
SNVS - Status, Not in Valid State, lower order cross-connections are still existing	Substructure deletion not allowed with LO cross-connections existing	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
SNVS - Status, Not in Valid State, lower order test access sessions are still existing	Substructure deletion not allowed with LO test access sessions existing	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
SNVS - Status, Not in Valid State, low-order crossconnection loopbacks not allowed with LOXC/1	No low-order crossconnection loopbacks with LOXC/1	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
SNVS - Status, Not in Valid State, low-order test-access sessions not allowed with LOXC/1	No low-order TACC sessions with LOXC/1	"CONN-TACC: Connect test access" (p. 3-94)
SNVS - Status, Not in Valid State, low-order transmission entities present	protection unit is active and LO transmission entities	"DLT-PROTN-GRP: Delete protection group" (p. 3-233)
SNVS - Status, Not in Valid State, LOXC destination not usable	No forced switch to unusable LOXC slot (LOXC Protection Group)	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, LOXC switch capacity invalid	LOXC pack with more than 0G switch capacity	"CONN-TACC: Connect test access" (p. 3-94); "ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, LOXC switch capacity invalid	LOXC pack with more than 0G switch capacity for conversion cross-connections	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, LOXC switch capacity invalid	LOXC switch capacity and unit roles	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, LOXC switch capacity invalid	No low-order loopbacks with LOXC cap. equal 0	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
SNVS - Status, Not in Valid State, Manual MCN routes still exist	Assignment to SCN/UNI/ENNI Node while manual MCN routes exist;	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, Manual MCN routes still exist	Check if manual MCN routes exist;	"ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Manual MCN routes still exist	IP disabling while manual MCN routes exist;	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, Manual routes exist	(UNI/E-NNI specific): Can't disable DCC if associated node has manual routes via this DCC	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Status, Not in Valid State, Manual routes exist	Can't de-assign LAN from a node if manual routes exist	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, Manual Switch for Protection Type 1+1_Optimized not allowed	MSP type 1+1_optimized does not allow manual switch commands	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, Manual Switch to Worker for Protection Type 1x1 not allowed	MSP type 1:1 does not allow manual switch requests to worker	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, Modification not allowed	No modification of rptmode if vlantagmd = 802.1TAG	"ED-EPORT: Edit Ethernet port" (p. 3-329)
SNVS - Status, Not in Valid State, Modification not allowed in Backward Compatible Mode	Parameter modification not allowed in backward compatible mode	"ED-EPORT: Edit Ethernet port" (p. 3-329); "ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, Modification of flow control mode not allowed	No Modification of fcmd if vlantagmd = 802.1TAG	"ED-EPORT: Edit Ethernet port" (p. 3-329)
SNVS - Status, Not in Valid State, multiple added XCs for MS-SPRing TOP	no multiple added XCs for MS-SPRing TOP	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, multiple added XCs for MS-SPRing TOP	No multiple added XCs for MS-SPRing TOP	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, multiple added XCs for MS-SPRing TOP	no multiple added XCs for TOP	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, ndetmd must not be AUTO	Some parameters not settable for ports of ndetmd AUTO	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, new Tributary has existing loopback	No cross-connection loopback existing	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing	(ONNS specific): Modify of TNA parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing	(ONNS specific): Modify of TNA parameters on VCAT layer	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, NN Cross-connections associated to end-to-end network connections still existing	Modify of EDGE parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, no dcc channel enabled	dcc enabled	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, no dcc channel enabled	dcc enabled on protection port	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, no dcc channel with mode UITS enabled	dcc enabled on worker port	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, no matching log entry	matching log entry	"RTRV-LOG-NTFCN: Retrieve notification log" (p. 3-2464); "RTRV-LOG-PROTNSW: Retrieve log protection switch" (p. 3-2468)
SNVS - Status, Not in Valid State, No matching NN cross-connection existing on working tributary	NN connection of matching directionality must exist on working tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, no other LAN port with IP enabled for T-TD or FTTD available	Can't assign LAN to SCN Node when T-TD or FTTD is enabled and no other LAN port with IP configuration is available	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, no other LAN port with IP enabled for T-TD or FTTD available	Can't assign LAN to UNI Node or E-NNI Node when T-TD or FTTD is enabled and no other LAN port with IP configuration is available	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, No restore (RFR) with ES64U pack	No DB restore with ES64U pack;	"COPY-RFILE: Copy remote file" (p. 3-106)
SNVS - Status, Not in Valid State, No restore with ES64U pack	No DB restore with ES64U Pack	"CPY-MEM: Copy memory" (p. 3-114)
SNVS - Status, Not in Valid State, no SCN IP address provisioned	(SCN specific): Check for provisioned SCN IP address	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Status, Not in Valid State, no SCN IP address provisioned	Can't assign LAN to scnnode when no ASTN IP address is provisioned	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, no SCN IP address provisioned	Can't assign LAN to uninode or enninode when no ASTN IP address is provisioned	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, no SCN IP address provisioned	Can't assign LAN to uninode when no ASTN and UNI IP address is provisioned	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, no SCN/ENNI IP address provisioned	Can't assign LAN to enninode when no ASTN and E-NNI IP address is provisioned	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, no SwiSS capable HW	Compatibility xcsetup and HW configuration	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, no test access session on the test tributary	No test access session on the test tributary	"CHG-ACCMD: Change access mode" (p. 3-65); "DISC-TACC: Disconnect test access" (p. 3-129)
SNVS - Status, Not in Valid State, no XC reservations allowed for change of iNNI parameters	Setting of some INNI properties only possible, if no reservations wrt. M:N protecting connections existing	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, no XC reservations allowed for removing of srlgList entries	Removing of entries from srlgList not allowed if reservations wrt. M:N protecting connections exist	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, no XCs allowed for change of iNNI parameters	Check of INNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, no XCs allowed for removing of srlgList entries	Removing of entries from srlgList not allowed if NN cross-connections exist	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, node in use as RA destination node	Reject deletion when node is used as RA destination node by OSI node 1	"DLT-ULSDCC-L3: Delete a node ID other than node-1" (p. 3-265)
SNVS - Status, Not in Valid State, node not in idle state	Idle ring node for protocol conversion	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, Not allowed for 1 GE interfaces	Not allowed for 1 GE interfaces	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, Not allowed for SDH interface standard	Not allowed for SDH	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Not allowed if VCAT disabled	Enabling of inmbs and outmbs not allowed while vcat disabled	"ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
SNVS - Status, Not in Valid State, Not allowed if VCAT disabled	Not allowed if virtual concatenation mode disabled	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, not in Maintenance Condition	Check of Maintenance Condition	"SET-SID: Set SID" (p. 3-3154)
SNVS - Status, Not in Valid State, not in Maintenance Condition	check of the maintenance condition	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, not in Maintenance Condition	Check whether system is in maintenance condition	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188); "ED-EC1: Edit EC1 port" (p. 3-317); "ED-EQPT (socket): Edit equipment" (p. 3-379); "ED-T3: Edit DS3 port" (p. 3-1202)
SNVS - Status, Not in Valid State, not in Maintenance Condition	Check whether system is maintenance condition	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "ED-EQPT (slot): Edit equipment (slot)" (p. 3-370); "ENT-EQPT (slot): Enter equipment (slot)" (p. 3-1547); "ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
SNVS - Status, Not in Valid State, not in Maintenance Condition	Command denied based on Maintenance Condition (SNVS)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SNVS - Status, Not in Valid State, Not in maintenance condition	check of the maintenance condition	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
SNVS - Status, Not in Valid State, Not in maintenance condition	check of the maintenance condition for SC IP address and/or SC ID change.	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
SNVS - Status, Not in Valid State, Not in maintenance condition	check of the maintenance condition for UNI IP address change	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
SNVS - Status, Not in Valid State, Not in Maintenance Condition	check of the maintenance condition	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
SNVS - Status, Not in Valid State, NVM contains DB for not existing ES64U packs	NVM contains DB for NOT existing ES64U pack;	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, NVM contains no DB for existing ES64U packs	NVM NOT contains DB for existing ES64U packs;	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SNVS - Status, Not in Valid State, NVM size not sufficient	Compatibility xcsetup and NVM size	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, one port of protection unit is assigned as timing reference	Working port to be added is assigned as timing reference	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, only allowed for EDGE port in lockout sate	No modification of trbmd at ports of portClass EDGEwhile portControl is not LOCKOUT	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, only allowed for EDGE port in lockout state	(ONNS specific): Modify of TNA parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, only allowed for EDGE port in lockout state	(ONNS specific): Modify of TNA parameters on VCAT layer	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, only allowed for EDGE port in lockout state	Modify of EDGE parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, only allowed for ENNI port in lockout state	Modify of ENNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, only allowed for ENNI port in lockout state`	No modification of trbmd at ports of portClass ENNI while portControl is not LOCKOUT	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
`SNVS - Status, Not in Valid State, only allowed for INNI port in lockout state`	Setting of INNI properties only possible if portControl is LOCKOUT	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, only allowed for UNI port in lockout state`	Modify of UNI parameters	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, ONNS is active`	Deletion of XC circuit packs while ONNS is active	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
`SNVS - Status, Not in Valid State, ONNS is active`	SW installation in trial phase while ONNS is active	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SNVS - Status, Not in Valid State, Optical Channel must be enabled`	Enabling of FEC only while Optical Channel is enabled	"ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Optical Channel must be enabled`	IDNV - Input, Data Not Valid, Parameter not supported	"ED-OC192: Edit OC192 port" (p. 3-663)
`SNVS - Status, Not in Valid State, optical wavelength is not provisioned`	Output can only be enabled when wavelength is provisioned	"ED-OTU2: Edit OTU2 port" (p. 3-730)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, osinode not in created state	osinode must be in the created state	"DLT-ULSDCC-L3: Delete a node ID other than node-1" (p. 3-265); "ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack" (p. 3-1312)
SNVS - Status, Not in Valid State, other port of protection unit is assigned to a line timing reference	no timing reference on other port of protection unit	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, OUC to different protection tributary already exists	No OUC must already exist on working tributary to different protection tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, Output is already crossconnected	Output not in use	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, parameter required	Parameter sc or dest is missing	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
SNVS - Status, Not in Valid State, Parameter value for SNC Type not compliant to interface standard of working and protection tributaries	SNC Type must comply with interface standard of affected EDGE ports	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, Parameter(s) present which are not allowed	Only those parameter present which are needed	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, P-bit parity violation monitoring not allowed with Unframed Clear-Channel	P-bit parity violation monitoring not allowed with unframed clear-channel	"ED-T3: Edit DS3 port" (p. 3-1202)
SNVS - Status, Not in Valid State, Port Class must be TRADITIONAL	(ONNS specific): Change the portClass away from EDGE	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, Port Class must not be EDGE or INNI	(ONNS specific): Restrict number of crossconnection using same input tributary on EDGE port	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, port is already used in another protection group	port in one group	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, Port is not out-of-service	Port is out-of-service	"OPR-LPBK: Operate facility loopback" (p. 3-2044)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, port is out of service`	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, port is out of service`	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, port is out of service`	Setting of portClass to INNI only possible, if service condition not OUTOFSERVICE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port is still in use`	Change the eportClass away from UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Port is still in use`	Change the portClass away from ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Port is still in use	Change the portClass away from UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port is still in use	Setting of portClass away from INNI only possible, if the port is not in use	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port is still in use	Setting of portClass away from INNI only possible, if the port is not in use by M:N reservation	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port is still in use	Setting of portClass from EDGE to TRADITIONAL only possible, if the tributaries on the port are not in use by ONNS:	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Port is still in use by ONNS`	Check if related port are not in use by ONNS	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
`SNVS - Status, Not in Valid State, Port is still in use by ONNS`	Check if related ports are not in use by ONNS	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
`SNVS - Status, Not in Valid State, Port is still in use by ONNS`	No port on protection pack in portClass EDGE	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, Port is still in use by ONNS`	No port on protection pack in portClass UNI	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, port mode change not allowed for INNI ports`	Port Mode not settable for ports of portClass = INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode`	(ONNS specific): Change the eportClass away from EDGE	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode`	Change the eportClass away from UNI;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode`	Change the portClass away from EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode	Change the portClass away from ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode	Change the portClass away from INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode	Change the portClass away from UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, Port must be in LOCKOUT mode	Modify of UNI parameters	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, Port must not be a 1:1 protection port	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Port must not be a 1:1 protection port`	Setting of portClass from TRADITIONAL to EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must not be a 1+1 protection port`	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
`SNVS - Status, Not in Valid State, Port must not be a 1+1 protection port`	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must not be a 1+1 protection port`	Setting of portClass from TRADITIONAL to EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must not be a 4f-BLSR protection port`	Setting of portClass from TRADITIONAL to EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Port must not be a member of a 1:1 protection group`	Change the portClass to ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must not be a member of a BLSR/MS-SPRing protection group`	Change the portClass away from UNI	"ED-STM1E: Edit STM1E port" (p. 3-870)
`SNVS - Status, Not in Valid State, Port must not be member of a protection group`	Setting of portClass to INNI not possible for ports in Protection Groups	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, Port must not be of portClass ENNI`	No loopbacks on tributaries below ports of portClass ENNI	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, Port must not be of portClass INNI`	No loopbacks on ports of portClass INNI	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
`SNVS - Status, Not in Valid State, Port must not be of portClass INNI`	No loopbacks on tributaries below ports of portClass INNI	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, Port must not be of portClass UNI`	No loopbacks on ports of portClass UNI	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
`SNVS - Status, Not in Valid State, Port must not be of portClass UNI`	No loopbacks on tributaries below ports of portClass UNI	"OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, port not in use for ENNI`	Port not used for ENNI	"DLT-ENNIMANROUTE: Delete manual route from the E-NNI node" (p. 3-164)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, port numbers cannot be changed when T-TD is enabled`	no TCP port number change with enabled T-TD function	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
`SNVS - Status, Not in Valid State, port rate not supported for this protection type`	supported port rate	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, port used for Linear MSP`	Port used for Linear MSP	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, port used for MS-SPRing/ BLSR`	Port used for MS-SPRing/BLSR	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, port used for NN application`	Port used for ONNS application	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, port(s) for OUC in adaptive mode`	Tributaries must not belong to ports of tributary mode ADAPTIVE	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, portClass may not be EDGE`	no EDGE protection port for 1:1 MSP	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass may not be EDGE`	no EDGE protection port for 1+1 MSP	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass may not be UNI`	no UNI protection port for 1:1 MSP	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass may not be UNI`	no UNI protection port for 1+1 MSP	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass must not be EDGE for TOP`	supported portClass for TOP	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass must not be ENNI`	Some parameter values not settable for ports of portClass ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, portClass must not be INNI`	(ONNS specific): hptimmon changes to ENABLE only allowed when corresponding port is not of portClass INNI	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
`SNVS - Status, Not in Valid State, portClass must not be INNI`	No port in portClass INNI	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, portClass must not be INNI`	portClass not INNI	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, portClass must not be INNI`	Some parameters not settable for ports of portClass = INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, portClass must not be INNI`	supported portClass	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass must not be UNI`	Some parameter values not settable for ports of portClass UNI	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, portClass must not be UNI`	Some parameters not settable for ports of portClass = UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
`SNVS - Status, Not in Valid State, portClass must not be UNI`	supported portClass	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, portClass of protection port must not be EDGE`	supported portClass	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, ports have different interface types`	common interface standard	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, ports have different interface types`	common interface type	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, ports have different port rates`	common port rate	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, ports must be on different circuit packs`	port on different packs	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, ports must be on different circuit packs	port on same pack	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, protection dcc channel is enabled	dcc protection side is enabled	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, protection group still exist	No protection groups allowed	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, protection port is assigned as timing reference	Timing Reference Assigned on Protection Unit	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
SNVS - Status, Not in Valid State, protection port is assigned to a line timing reference	no timing reference on protection port	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, protection port is assigned to a line timing reference	Timing reference on protection port only allowed in asymmetric spans of a ring	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, protection port is out-of-service	no out-of-service state on protection port	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, Protection switch with current combination of circuit pack types is traffic affecting	No manual protection switch if the switchover leads to traffic impairments	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, protection unit is active	protection unit is active and transmission entities exist	"DLT-PROTN-GRP: Delete protection group" (p. 3-233)
SNVS - Status, Not in Valid State, RC neighbor still exists	Neighboring routing controller still exists	"CANC-L1ROUTING" (p. 3-54)
SNVS - Status, Not in Valid State, reference unavailable	Reference switching during lockout or disabled state	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
SNVS - Status, Not in Valid State, request not practicable	Forced or manual switch request while reference is in forced state	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
SNVS - Status, Not in Valid State, request not practicable	No change of the request state, request not practicable	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
SNVS - Status, Not in Valid State, resources in use by SwiSS	Setting xcsetup back to TRADITIONAL	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, ring circuit map discovery not yet completed	Invalid ring map	"RTRV-CKT: Retrieve circuit" (p. 3-2274)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Roll of low-order cross-connections not allowed with LOXC/ 1	No Rolling of low-order cross-connections with LOXC/1	"ENT-ROLL: Enter roll" (p. 3-1796)
SNVS - Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port	inputsig consistent with existing cross-connections	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port	inputsig consistent with existing cross-connections for fixed tributary mode	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port	unequiposig consistent with existing cross-connections	"ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, signal structure cannot be changed for connected parts of a fixed port	unequiposig consistent with existing cross-connections for fixed tributary mode	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585)
SNVS - Status, Not in Valid State, socket is not provisioned	Socket must already be provisioned	"ED-EQPT (socket): Edit equipment" (p. 3-379)
SNVS - Status, Not in Valid State, socket is not provisioned	Socket must be provisioned	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, Software generic corrupt or not available	APPLY applied on corrupted software	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SNVS - Status, Not in Valid State, SONET port not allowed for TOP	SDH ports for TOP	"ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, static VLAN configuration not cleared	Check whether VLAN lists are empty	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SNVS - Status, Not in Valid State, SW generic is in trial phase	Compatibility xcsetup and software state	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, SW generic is in trial phase	No SW download, DB download or DB restore during trial phase	"CPY-MEM: Copy memory" (p. 3-114)
SNVS - Status, Not in Valid State, SW generic is in trial phase	No SW download, DB download, or DB restore during trial phase	"COPY-RFILE: Copy remote file" (p. 3-106)
SNVS - Status, Not in Valid State, SwiSS access possible	(ONNS specific): No interface standard change when in use by ONNS	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, SwiSS access possible	No ONNS for interface standard change	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, SwiSS access possible	No ONNS for interface standard change on GE10PL1 pack	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SNVS - Status, Not in Valid State, SwiSS not enabled	(SCN specific): Check whether ONNS is enabled	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SNVS - Status, Not in Valid State, SwiSS not enabled	Do not accept ONNS TL1 commands when xcsetup = TRADITIONAL	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, SwiSS not enabled	Do not accept setting portclassdflt to EDGE when xcsetup = TRADITIONAL	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
SNVS - Status, Not in Valid State, SwiSS not enabled	Modification of SCN related parameters only while ONNS is enabled	"ED-NE: Edit NE" (p. 3-442)
SNVS - Status, Not in Valid State, SwiSS not enabled	portClass only settable when xcsetup = SWISS	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, switch back to previous software only allowed in trial phase	Switch back to previous SW only during trial phase	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, switch capacity below provisioned XC CP switch capacity`	Maximum switch capacity according XC CP	"ED-EQPT (system): Edit equipment (system)" (p. 3-391)
`SNVS - Status, Not in Valid State, switch command not allowed for actual protection state`	allowed switch command in that state	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
`SNVS - Status, Not in Valid State, system initializing`	System Initializing not completed (SNVS)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SNVS - Status, Not in Valid State, target Port Class must be TRADITIONAL`	Change the eportClass away from UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, target Port Class must be TRADITIONAL`	Change the portClass away from EDGE	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, target Port Class must be TRADITIONAL`	Change the portClass away from ENNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, target Port Class must be TRADITIONAL`	Change the portClass away from INNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, target Port Class must be TRADITIONAL	Change the portClass away from UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, TARP not enabled	Check for correct TARP configuration	"ED-NE: Edit NE" (p. 3-442)
SNVS - Status, Not in Valid State, TARP origination function disabled	TARP data cache change only if TARP origination enabled	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
SNVS - Status, Not in Valid State, TCA profile assigned	deleting assigned profile	"DLT-TCA-PROF: Delete TCA profile" (p. 3-259)
SNVS - Status, Not in Valid State, Test Access not allowed with path protected tributaries	Not allowed for path protected tributaries;	"CONN-TACC: Connect test access" (p. 3-94)
SNVS - Status, Not in Valid State, test access session exists	no test access session for istd change	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
SNVS - Status, Not in Valid State, the CP is not plugged out	CP must be plugged out	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177)
SNVS - Status, Not in Valid State, the module is not plugged out	Exchangeability II: Invalid module plugged in	"ED-EQPT (socket): Edit equipment" (p. 3-379)
SNVS - Status, Not in Valid State, the module is not plugged out	Module must be plugged out	"ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
SNVS - Status, Not in Valid State, the Module is not plugged out	module must be plugged out	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, TID inconsistent with existing cross-connection	TIDs consistent with existing cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, timing clock mode inconsistent	Timing clock mode inconsistent	"SET-SYNCN: Set synchronization" (p. 3-3158)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, Timing reference is assigned to this port	servcond must be enabled when timing source	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM1E: Edit STM1E port" (p. 3-870); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
SNVS - Status, Not in Valid State, TMUX destination not usable	No forced switch to unusable TransMUX pack (TransMUX Protection Group);	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SNVS - Status, Not in Valid State, TMUX switch capacity invalid	TransMUX switch capacity and unit roles	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, Too many crossconnection originating from same tributary	(ONNS specific): Restrict number of crossconnection using same input tributary on EDGE port	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, Too many crossconnection originating from same tributary	Restrict number of cross-connection using same input tributary	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, Too many cross-connection originating from same tributary	Restrict number of cross-connection using same input tributary	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Transparent DCC connections are existing at protection pack	Transparent DCC connections	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, transparent DCC connections are still existing	No transparent DCC connections	"DLT-EQPT (slot): Delete equipment (slot)" (p. 3-177); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
SNVS - Status, Not in Valid State, Trial to change loopback type	No change of loopback allowed	"OPR-LPBK: Operate facility loopback" (p. 3-2044)
SNVS - Status, Not in Valid State, trib of 1x1 protection port as input-trib in pathprotected crossconnection	No trib of any MSP 1x1 protection port as in_aid for HO pathprotected cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, trib of 1x1 protection port as input-trib in pathprotected cross-connection	No trib of any MSP 1x1 protection port as newinaid for HO pathprotected cross-connection;	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO crossconnection	No input tributary of LO pathprotected cross-connection on the worker port	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO crossconnection	No trib of any MSP port as in_aid for LO path-protected cross-connection	"ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, trib of MSP 1+1/1x1 port as input-trib in pathprotected LO cross-connection	No trib of any MSP port as newinaid for LO pathprotected cross-connection;	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, Tributaries must be on ports of portClass EDGE	Tributaries must belong to ports of portClass EDGE	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI	(ONNS specific): LO Tributaries must not belong to ports of portClass ENNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Tributaries must not be on ports of portClass ENNI	(ONNS specific): Tributaries must not belong to ports of portClass ENNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Tributaries must not be on ports of portClass INNI	(ONNS specific): Tributaries must not belong to ports of portClass INNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Tributaries must not be on ports of portClass UNI	(ONNS specific): LO Tributaries must not belong to ports of portClass UNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, Tributaries must not be on ports of portClass UNI	(ONNS specific): Tributaries must not belong to ports of portClass UNI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SNVS - Status, Not in Valid State, tributaries used in test access session	Tributaries not used in a test access session	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, tributaries used in test access session	Tributaries not used in a test access session when removing a path protection	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
SNVS - Status, Not in Valid State, Tributary at port in adaptive mode	LO Substructure creation not allowed on ports in adaptive mode	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
SNVS - Status, Not in Valid State, Tributary has existing loopback	LO Substructure creation not allowed while loopback exists	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Tributary has existing loopback`	No cross-connection loopback existing	"DLT-CRS: Delete cross-connection" (p. 3-145); "ENT-CRS: Enter cross-connection" (p. 3-1464)
`SNVS - Status, Not in Valid State, Tributary has existing loopback`	Test tributary or E or F tributary in a loopback	"CONN-TACC: Connect test access" (p. 3-94)
`SNVS - Status, Not in Valid State, Tributary is not in state wait-to-restore`	wait-to-restore is not pending	"OPR-VCGTRIB: Operate virtual concatenation group tributary" (p. 3-2082)
`SNVS - Status, Not in Valid State, Tributary is used in cross-connection`	no XC loopback on tribs	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, Tributary is used in cross-connection loopback`	no XC loopback on tribs	"DLT-PROTN-GRP: Delete protection group" (p. 3-233)
`SNVS - Status, Not in Valid State, Tributary is used in cross-connection loopback`	No XC loopbacks on protection tribs	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, Tributary is used in cross-connection loopback`	no XC loopbacks on tribs	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, Tributary not assigned as a test tributary`	Tributary is not assigned as a test tributary	"RTRV-TACC: Retrieve test access" (p. 3-2974)
`SNVS - Status, Not in Valid State, Tributary used in test access session`	LO Substructure creation not allowed while test session exists	"ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary." (p. 3-1191)
`SNVS - Status, Not in Valid State, Tributary used in test access session`	No test access session	"CONN-TACC: Connect test access" (p. 3-94); "DLT-CRS: Delete cross-connection" (p. 3-145); "DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188); "ENT-CRS: Enter cross-connection" (p. 3-1464); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745); "OPR-LPBK: Operate cross-connection loopback" (p. 3-2035)
`SNVS - Status, Not in Valid State, Tributary used in test access session`	no test access session active	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, T-TD enabled`	Not allowed to disable TARP when T-TD is enabled	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, Unframed Clear-Channel not allowed with DS3 path PM`	unframed clear-channel not allowed with DS3 path PM	"ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, Unframed Clear-Channel not allowed with P-bit parity violation monitoring`	unframed clear-channel format not allowed with P-bit parity violation monitoring	"ED-T3: Edit DS3 port" (p. 3-1202)
`SNVS - Status, Not in Valid State, UNI Node IP address not provisioned`	UNI node IP Address must be provisioned	"ED-RSVP: Modify RSVP parameters" (p. 3-857)
`SNVS - Status, Not in Valid State, UNI Node IP Address not provisioned`	UNI node must be created	"RTRV-RSVP: Retrieve RSVP parameters" (p. 3-2823)
`SNVS - Status, Not in Valid State, UNI not enabeld`	Change the portClass to UNI	"ED-STM1E: Edit STM1E port" (p. 3-870)
`SNVS - Status, Not in Valid State, UNI not enabled`	(ONNS specific): Change the eportClass to UNI	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, UNI not enabled`	Change the eportClass to UNI;	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, UNI not enabled`	Change the portClass to UNI	"ED-OC12: Edit OC12 port" (p. 3-523); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC3: Edit OC3 port" (p. 3-461); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-STM1: Edit STM1 port" (p. 3-915); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM4: Edit STM4 port" (p. 3-975); "ED-STM64: Edit STM64 port" (p. 3-1111)
`SNVS - Status, Not in Valid State, unknown IP address within MAP`	Check for known IP address	"DLT-IP-MAP: Delete TCP/IP map" (p. 3-198)
`SNVS - Status, Not in Valid State, unknown module equipage`	Equipage status and inventory data of module must be accessible	"DLT-EQPT (socket): Delete equipment (socket)" (p. 3-188)
`SNVS - Status, Not in Valid State, unknown module equipage`	Equipage Status and Inventory Data of Module must be accessible	"ENT-EQPT (socket): Enter equipment (socket)" (p. 3-1557)
`SNVS - Status, Not in Valid State, unknown module equipage`	Exchangeability I: Equipage status and inventory data of module must be accessible	"ED-EQPT (socket): Edit equipment" (p. 3-379)
`SNVS - Status, Not in Valid State, unknown TARP Manual Adjacent NSAP`	Check for existing TARP manual adjacent NSAP	"DLT-ULSDCC-L4: Delete TARP manual adjacent NSAP address in layer 4 of the specified OSI stack" (p. 3-269)

Error code	Meaning (validation rule)	Related to command
`SNVS - Status, Not in Valid State, unsupported MS-SPRing/BLSR DNI/DRI`	Supported MS-SPRing/BLSR DNI/DRI	"CNVT-CCT: Convert cross-connection topology" (p. 3-73)
`SNVS - Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`	No configuration of unsupported VCG tributary structure for EDGE 10GEPL VCG	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`	No configuration of unsupported VCG tributary structure for EDGE or UNI 1GEPL VCG	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`	Setting of eportClass of a 10GEPL VCG to EDGE only possible, if the configured tributary structure is supported	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`	Setting of eportClass of a 1GEPL VCG to EDGE only possible, if the configured tributary structure is supported	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Unsupported VCG tributary structure for actual eportClass`	Setting of eportClass of a 1GEPL VCG to UNI only possible, if the configured tributary structure is supported	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, VCG is still in use`	(ONNS specific): Setting of eportClass from EDGE to TRADITIONAL only possible, if the VCG tributaries on the port are not in use by ONNS	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, VCG members are still enabled`	Not allowed if a member is enabled	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, Virtual concatenation is disabled`	Parameter modification restrictions while vcat is set to DISABLE	"ED-VCG: Edit virtual concatenation group" (p. 3-1354)
`SNVS - Status, Not in Valid State, worker and protection LOXC must be of same type`	Worker and protection LOXC must be of same type	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
`SNVS - Status, Not in Valid State, working dcc channel is disabled`	dcc protection: working side is disabled	"ED-PROTN-GRP: Edit protection group" (p. 3-803)
`SNVS - Status, Not in Valid State, wrong IP configuration`	check for correct OSI node configuration	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
`SNVS - Status, Not in Valid State, wrong IP configuration`	Check for IP default router LAN	"ENT-FECOM-LAN: Enter LAN related parameters" (p. 3-1581)
`SNVS - Status, Not in Valid State, XC destination not usable`	No forced switch to unusable XC slot (Switching Protection Group)	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
`SNVS - Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary`	No forced XC loopback must exist on EDGE-prt tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
`SNVS - Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary`	No forced XC loopback must exist on EDGE-wrk or EDGE-prt tributary	"DLT-OUC: Delete ONNS to UPSR/SNCP constructs" (p. 3-217)

Error code	Meaning (validation rule)	Related to command
SNVS - Status, Not in Valid State, XC loopback exists on EDGE-wrk or EDGE-prt tributary	No forced XC loopback must exist on EDGE-wrk tributary	"ENT-OUC: Enter ONNS to UPSR/SNCP constructs" (p. 3-1691)
SNVS - Status, Not in Valid State, XC loopbacks still exist at the protection pack	No XC loopbacks allowed	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, XC Pack not provisioned	XC pack is (pre-)provisioned	"CONN-TACC: Connect test access" (p. 3-94); "ENT-CRS: Enter cross-connection" (p. 3-1464); "ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SNVS - Status, Not in Valid State, xcsetup must be TRADITIONAL	check of xcsetup for ip_ad_astn change	"ENT-ULS: Enter upper layer stack" (p. 3-1944)
SNVS - Status, Not in Valid State, tributary used in test access session	no test access session active	"DLT-PROTN-GRP: Delete protection group" (p. 3-233)
SNVS - Status, Not in Valid State. Duplex Control Recovery in progress	No SW installation during Duplex Control Recovery	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SNVS - Status, Not Valid State	Automatic ring discovery to be disabled	"ED-FETID: Edit far end target identifier" (p. 3-405); "SET-MAP-RING: Set map ring" (p. 3-3142)
SPFA - Status, Protection unit FAiled	RTRV-SPANSW needs functioning XC pack	"RTRV-SPANSWS: Retrieve SPAN switch state" (p. 3-2830)
SPFA - Status, Protection unit FAiled	succeeding switch command	"OPR-PROTNSW: Operate protection switch" (p. 3-2057); "RLS-PROTNSW: Release protection switch" (p. 3-2203)
SPFA - Status, Protection unit FAiled, APS did not succeed	APS switch time constraint	"OPR-PROTNSW: Operate protection switch" (p. 3-2057)
SRAC - Status, Requested Access Configuration invalid, retrieved file NE mismatch	Product mismatch	"CPY-MEM: Copy memory" (p. 3-114)
SRAC - Status, Requested Access Configuration is invalid, retrieved file NE mismatch	Product Mismatch (SRAC)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SROF - Status, Requested Operation Failed	no more than two TARP manual adjacencies per node	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
SROF - Status, Requested Operation Failed	Non-Hardware system difficulties	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
SROF - Status, Requested Operation Failed	Operation Failed (SROF)	"RTRV-HDR: Retrieve header" (p. 3-2417)

Error code	Meaning (validation rule)	Related to command
`SROF - Status, Requested Operation Failed`	Provisioning of TARP manual adjacencies only if TARP enabled	"ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks " (p. 3-1961)
`SROF - Status, Requested Operation Failed, external communications failure`	Communication Failure (SROF)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SROF - Status, Requested Operation Failed, external communications failure`	External communication failure	"CPY-MEM: Copy memory" (p. 3-114)
`SROF - Status, Requested Operation Failed, invalid ES64U switch capacity without XC`	No ES64U capacity setting without XC;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SROF - Status, Requested Operation Failed, not all ES64U databases preserved`	Not all databases of ES64U preserved for trial phase;	"APPLY: Apply software generic or activate a downloaded database" (p. 3-35)
`SROF - Status, Requested Operation Failed, processing failure`	Processing Failed (SROF)	"RTRV-HDR: Retrieve header" (p. 3-2417)
`SROF - Status, Requested Operation Failed, unsuccessful file retransmission`	Backup failed after unsuccessful retransmission of files;	"CPY-MEM: Copy memory" (p. 3-114)
`SROF - Status, Requested Operation Failed,, invalid LOXC switch capacity without XC`	No LO switch capacity setting without XC	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SROF - Status, Requested Operation Failed,, invalid TransMUX switch capacity without XC`	No TransMUX capacity setting without XC	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SRQN - Status, invalid request, uid not logged in`	user with uid not logged in	"CANC-USER-SECU: Cancel user security" (p. 3-62)
`SSRD - Status, Switch Request Denied, clock mode inconsistent`	Clock backup switch request when clock mode is free running	"OPR-SYNCNSW: Operate synchronization switch " (p. 3-2077)
`SSRE - Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`	Maximum number of user sessions via T-TD on Gateway NE	"ACT-USER: Activate user" (p. 3-24)
`SSRE - Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`	maximum number of user sessions via T-TD on Gateway NE per TCP connection	"ACT-USER: Activate user" (p. 3-24)
`SSRE - Status, System Resources Exceeded (sent by T-TD, TL1_TTD)`	Maximum number of user sessions via T-TD on Gateway NE with CTL/4	"ACT-USER: Activate user" (p. 3-24)
`SSRE - Status, System Resources Exceeded,  no free DDC-M channel available for ES64U`	Check whether DCC Channel available for ES64U circuit pack ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
`SSRE - Status, System Resources Exceeded, allowed limit exceeded`	Internal Limit exceeded (SSRE)	"RTRV-HDR: Retrieve header" (p. 3-2417)

Error code	Meaning (validation rule)	Related to command
SSRE - Status, System Resources Exceeded, allowed limit of DCC channels exceeded	(UNI/E-NNI specific): Maximum number of channels assigned to E-NNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, allowed limit of DCC channels exceeded	(UNI/E-NNI specific): Maximum number of channels assigned to UNI	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, allowed limit of neighboring RCs exceeded	Maximum Number of Neighboring RCs	"ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers" (p. 3-1792)
SSRE - Status, System Resources Exceeded, allowed limit of SCN DCC channels exceeded	(SCN specific): Maximum number of channels assigned to SCN	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, allowed number of active ES64U packs exceeded	Check number of active ES64U packs ;	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SSRE - Status, System Resources Exceeded, allowed number of active LOXC packs exceeded	Check number of active LOXC packs	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SSRE - Status, System Resources Exceeded, allowed number of active TransMUX packs exceeded	Check number of active TransMUX packs	"ED-EQPT (CP): Edit equipment (circuit pack)" (p. 3-349)
SSRE - Status, System Resources Exceeded, allowed number of MS SPRING and BLSR protection groups is exceeded	maximum number of 2-fibre and 4-fibre MS SPRING / BLSR protection groups	"ENT-PROTN-GRP: Enter protection group" (p. 3-1745)
SSRE - Status, System Resources Exceeded, ENNI limit of manual route entries exceeded	Check if limit of 64 Manual Routes per ENNI Node exceeded	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528)
SSRE - Status, System Resources Exceeded, limit of IP-MAP entries exceeded	Check if limit of 20 IP MAP entries exceeded	"ENT-IP-MAP: Enter TCP/IP map" (p. 3-1594)
SSRE - Status, System Resources Exceeded, limit of manual route entries exceeded	Check if limit of 20 Manual MCN IP Routes exceeded	"ENT-MANROUTE-MCN: Enter manual IP route for management communication" (p. 3-1614)

Error code	Meaning (validation rule)	Related to command
SSRE - Status, System Resources Exceeded, limit of manual route entries exceeded	Check if limit of 20 Manual Routes per LAN port exceeded	"DLT-ENNIMANROUTE notification" (p. 3-168); "ENT-ENNIMANROUTE: Enter manual route to the E-NNI node" (p. 3-1520); "ENT-ENNIMANROUTE notification" (p. 3-1528); "ENT-MANROUTE-LAN: Enter manual route at LAN port" (p. 3-1610)
SSRE - Status, System Resources Exceeded, limit of PJE counters exceeded	Not more than one PJE PM monitor per port	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
SSRE - Status, System Resources Exceeded, limit of PM monitors exceeded	maximum number of enabled PM monitors	"ED-T3: Edit DS3 port" (p. 3-1202)
SSRE - Status, System Resources Exceeded, limit of PM monitors exceeded	Maximum number of enabled PM monitors	"ED-EC1: Edit EC1 port" (p. 3-317); "ED-EPORT: Edit Ethernet port" (p. 3-329); "ED-OC192: Edit OC192 port" (p. 3-663); "ED-OC48: Edit OC48 port" (p. 3-585); "ED-OC48T: Edit OC48 transparent port" (p. 3-648); "ED-OTU2: Edit OTU2 port" (p. 3-730); "ED-PTF: Edit tributary with Path Termination Function (PTF)" (p. 3-830); "ED-STM16: Edit STM16 port" (p. 3-1035); "ED-STM16T: Edit STM16T port" (p. 3-1097); "ED-STM64: Edit STM64 port" (p. 3-1111); "ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339); "ED-VCG: Edit virtual concatenation group" (p. 3-1354); "ED-VCGTRIB: Edit virtual concatenation group tributary" (p. 3-1416)
SSRE - Status, System Resources Exceeded, limit of ppgnames exceeded	Maximum number of ppgnames exceeded	"CNVT-CCT: Convert cross-connection topology" (p. 3-73); "ED-PROTN-GRP: Edit protection group" (p. 3-803); "ENT-CRS: Enter cross-connection" (p. 3-1464)
SSRE - Status, System Resources Exceeded, limit of profiles exceeded	Maximum number of profiles	"ENT-ASAP-PROF: Enter alarm severity assignment profile" (p. 3-1446)
SSRE - Status, System Resources Exceeded, limit of profiles exceeded	Maximum number of TCA profiles reached	"ENT-TCA-PROF: Enter TCA Profile" (p. 3-1894)

Error code	Meaning (validation rule)	Related to command
SSRE - Status, System Resources Exceeded, limit of transparent DCC cross-connections exceeded	Maximum of DCC CCs exceeded with CTL/- or CTL/2	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
SSRE - Status, System Resources Exceeded, limit of transparent DCC cross-connections exceeded	Maximum of DCC CCs exceeded with CTL/3;	"ENT-CRS-FECOM: Enter transparent DCC cross-connection" (p. 3-1510)
SSRE - Status, System Resources Exceeded, limit of tribs with loca, locz exceeded	(ONNS specific): Maximum number of tributaries with loca, locz entries must not exceed 100	"ED-STS1: Edit STS1 tributary" (p. 3-1177); "ED-VC3: Edit VC3 tributary" (p. 3-1339)
SSRE - Status, System Resources Exceeded, limit of VLAN Identifiers exceeded	Maximum number of entries in valid VLAN ID list exceeded	"ED-EPORT: Edit Ethernet port" (p. 3-329)
SSRE - Status, System Resources Exceeded, no sufficient backplane bandwidth available	Sufficient backplane bandwidth available	"ENT-VCG – Enter virtual concatenation group" (p. 3-1983)
SSRE - Status, System Resources Exceeded, overall limit of DCC channels exceeded	Maximum number of channels assigned to MCN on CTL/-	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, overall limit of DCC channels exceeded	Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL/2	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, overall limit of DCC channels exceeded	Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL/3;	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSRE - Status, System Resources Exceeded, overall limit of DCC channels exceeded	Maximum number of channels assigned to MCN, SCN, UNI, and E-NNI on CTL4	"ENT-FECOM: Enter DCC channel related parameters" (p. 3-1565)
SSTP - Status, Stopped, failed target hardware	Failed Target Hardware (SSTP)	"RTRV-HDR: Retrieve header" (p. 3-2417)
SWFA - Status, Working unit Failed, control hardware failed, missing or initializing	Hardware Failed (SWFA)	"RTRV-HDR: Retrieve header" (p. 3-2417)

15 User privilege codes

List of user privilege codes

TL1 command	User privilege code (UCFC/UCAL)
ACT-L1ROUTING	P1
ACT-USER: Activate user	Any
ALW-MSG: Allow message	M3
APPLY: Apply software generic or activate a downloaded database	S4
AUD: Perform audit data	P3
CANC-L1ROUTING	P1
CANC-USER: Cancel user	Any
CANC-USER-SECU: Cancel user security	S5
CHG-ACCMD: Change access mode	T4
CNVT-CCT: Convert cross-connection topology	P3
CONN-TACC: Connect test access	T4
COPY-RFILE: Copy remote file	S4
CPY-MEM: Copy memory	S4
DISC-TACC: Disconnect test access	T4
DLT-ASAP-PROF: Delete alarm severity assignment profile	M3
DLT-CONN: Delete network connection	P3
DLT-CRS: Delete cross-connection	P3
DLT-CRS-FECOM: Delete transparent DCC cross-connection	M3
DLT-ENNIMANROUTE: Delete manual route from the E-NNI node	S4
DLT-ENNIMANROUTE notification	S4
DLT-EQPT (slot): Delete equipment (slot)	P3
DLT-EQPT (socket): Delete equipment (socket)	P3
DLT-IP-MAP: Delete TCP/IP map	S3

TL1 command	User privilege code (UCFC/UCAL)
DLT-MANROUTE-LAN: Delete manual route at LAN port	S4
DLT-MANROUTE-MCN: Delete one manual MCN IP route entry	S4
DLT-OUC: Delete ONNS to UPSR/SNCP constructs	P3
DLT-PATH: Delete ONNS path connection	P3
DLT-PROTN-GRP: Delete protection group	M4
DLT-RC-NEIGHBOR: Delete a routing controller from the list of neighboring routing controllers	S3
DLT-TCA-PROF: Delete TCA profile	PM3
DLT-ULSDCC-L3: Delete a node ID other than node-1	S3
DLT-ULSDCC-L4: Delete TARP manual adjacent NSAP address in layer 4 of the specified OSI stack	S3
DLT-USER-SECU: Delete user security	S5
DLT-VCG – Enter virtual concatenation group	P3
ED-ASAP-PROF: Edit alarm severity assignment profile	M3
ED-CRS: Edit cross-connection	P3
ED-DAT: Edit date and time	S4
ED-EC1: Edit EC1 port	P3
ED-EPORT: Edit Ethernet port	P3
ED-EQPT (CP): Edit equipment (circuit pack)	P3
ED-EQPT (shelf): Edit equipment (shelf)	P3
ED-EQPT (slot): Edit equipment (slot)	P3
ED-EQPT (socket): Edit equipment	P3
ED-EQPT (system): Edit equipment (system)	P3
ED-FETID: Edit far end target identifier	S3
ED-LONIM: Edit lower-order non-intrusive monitor	P3
ED-LOPTF: Edit lower-order path termination function	P3
ED-MAP-RING: Edit map ring	M3
ED-NE: Edit NE	S3
ED-NE-SECU: Edit NE security	S5
ED-NNCNFG: Modify ONNS system configuration	P4
ED-NNCRS: Edit NN cross-connection	P5
ED-OC3: Edit OC3 port	P3
ED-OC12: Edit OC12 port	P3
ED-OC48: Edit OC48 port	P3
ED-OC48T: Edit OC48 transparent port	P3

TL1 command	User privilege code (UCFC/UCAL)
ED-OC192: Edit OC192 port	P3
ED-OTU2: Edit OTU2 port	P3
ED-PATH: Add or modify alias of ONNS connection	P3
ED-PATH-PROT: Modify ONNS connection parameters or protection type	P3
ED-PATH-ROLL: Modify ONNS connection route	P3
ED-PID: Edit password identifier	S1
ED-PROTN-GRP: Edit protection group	M4
ED-PTF: Edit tributary with Path Termination Function (PTF)	P3
ED-RC-GENERAL: Edit general routing controller settings	S3
ED-RSVP: Modify RSVP parameters	S3
ED-STATE-EQPT (system): Edit equipment state	M4
ED-STM1E: Edit STM1E port	P3
ED-STM1: Edit STM1 port	P3
ED-STM4: Edit STM4 port	P3
ED-STM16: Edit STM16 port	P3
ED-STM16T: Edit STM16T port	P3
ED-STM64: Edit STM64 port	P3
ED-STS1: Edit STS1 tributary	P3
ED-SUBS: Create or delete the lower-order substructre at a higher-order tributary.	P3
ED-SYNCDATA: Synchronization of ONNS data	P3
ED-T3: Edit DS3 port	P3
ED-TCA-PROF: Edit TCA profile	PM3
ED-TMAT3: Edit asynchronous adjunct port of the TransMUX card	P3
ED-ULSDCC-L3: Enter layer 3 related parameters of the OSI stack	S4
ED-USER: Edit user	S1
ED-USER-SECU: Edit user security	S5
ED-VC3: Edit VC3 tributary	P3
ED-VCG: Edit virtual concatenation group	P3
ED-VCG-TRANS: Edit virtual concatenation group for fully transparent application	P3
ED-VCGTRIB: Edit virtual concatenation group tributary	P3
ED-VRTSW: Edit virtual switch	P3
ENT-ASAP-PROF: Enter alarm severity assignment profile	M3
ENT-BANNER	S5

TL1 command	User privilege code (UCFC/UCAL)
ENT-CONN: Establish network connections with TNA address parameters	P3
ENT-CRS: Enter cross-connection	P3
ENT-CRS-FECOM: Enter transparent DCC cross-connection	M3
ENT-ENNIMANROUTE: Enter manual route to the E-NNI node	S4
ENT-ENNIMANROUTE notification	S4
ENT-EQPT (slot): Enter equipment (slot)	P3
ENT-EQPT (socket): Enter equipment (socket)	P3
ENT-FECOM: Enter DCC channel related parameters	S3
ENT-FECOM-LAN: Enter LAN related parameters	S4
ENT-IP-MAP: Enter TCP/IP map	S3
ENT-MANROUTE-LAN: Enter manual route at LAN port	S4
ENT-MANROUTE-MCN: Enter manual IP route for management communication	S4
ENT-OUC: Enter ONNS to UPSR/SNCP constructs	P3
ENT-PATH: Create ONNS path connection	P3
ENT-PROTN-GRP: Enter protection group	M4
ENT-RC-NEIGHBOR: Add a routing controller to the list of neighboring routing controllers	S3
ENT-ROLL: Enter roll	P3
ENT-TCA-PROF: Enter TCA Profile	PM3
ENT-ULS: Enter upper layer stack	S4
ENT-ULSDCC-L3: Create a further OSI node	S4
ENT-ULSDCC-L4: Enter layer 4 related parameters of the OSI stacks	S4
ENT-USER-SECU: Enter user security	S5
ENT-VCG – Enter virtual concatenation group	P3
INH-MSG: Inhibit message	M3
INIT-EQPT (slot): Initialize equipment (slot)	M4
INIT-SYS: Initialize system	S4
OPR-DEFRAG: Perform defragmentation	P3
OPR-EXT-CONT: Operate external control	M3
OPR-LPBK: Operate cross-connection loopback	T4
OPR-LPBK: Operate facility loopback	T4
OPR-MAINT-ONNS: Maintenance actions for the ONNS application	P5
OPR-PROTNSW: Operate protection switch	M4
OPR-SYNCNSW: Operate synchronization switch	M4

TL1 command	User privilege code (UCFC/UCAL)
OPR-VCGTRIB: Operate virtual concatenation group tributary	M3
RLS-EXT-CONT: Release external control	M3
RLS-LPBK: Release loopback (cross-connection)	T4
RLS-LPBK: Release loopback (facility)	T4
RLS-PROTNSW: Release protection switch	M4
RLS-SYNCNSW: Release synchronization switch	M4
RTRV-ACTL-CONT: Retrieve actual condition types	M1
RTRV-ACTLCP: Retrieve acceptable circuit pack list	P1
RTRV-ALL: Retrieve all (port)	P1
RTRV-ALL: Retrieve all (tributary)	P1
RTRV-ALM: Retrieve alarm	M1
RTRV-ALM-ENV: Retrieve alarm environment	M1
RTRV-AO: Retrieve autonomous output	S1
RTRV-ASAP-ASGNMT: Retrieve ASAP assignment	M1
RTRV-ASAP-PROF: Retrieve ASAP profile	M1
RTRV-ATTR-CONT: Retrieve attribute control	M1
RTRV-ATTR-ENV: Retrieve attribute environment	M1
RTRV-ATTR-MSG: Retrieve attribute message	M1
RTRV-AUDLOG: Retrieve audit result log for ONNS connection data	P1
RTRV-BANNER	S1
RTRV-CKT: Retrieve circuit	M1
RTRV-COND: Retrieve condition	M1
RTRV-CONN: Retrieve data relevant for network connections	P1
RTRV-CRS: Retrieve cross-connection	P1
RTRV-CRS-FECOM: Retrieve transparent DCC cross-connection	M1
RTRV-EC1: Retrieve EC1 port	P1
RTRV-ELDR: Retrieve ONNS external link data repository	P1
RTRV-ENNIMANROUTE: Retrieve all manual routes of the E-NNI node	S1
RTRV-EPM: Ethernet performance monitoring	PM1
RTRV-EPORT: Retrieve Ethernet port	P1
RTRV-EQPT (bay): Retrieve equipment (bay)	P1
RTRV-EQPT (CP): Retrieve equipment (circuit pack)	P1
RTRV-EQPT (shelf): Retrieve equipment (shelf)	P1
RTRV-EQPT (slot): Retrieve equipment (slot)	P1

TL1 command	User privilege code (UCFC/UCAL)
RTRV-EQPT (socket): Retrieve equipment (socket)	P1
RTRV-EQPT (system): Retrieve Equipment (system)	P1
RTRV-EXT-CONT: Retrieve external control	M1
RTRV-FECOM: Retrieve DCC related parameters	M1
RTRV-FECOM-LAN: Retrieve LAN channel related parameters	S1
RTRV-HDR: Retrieve header	Any
RTRV-IP-MAP: Retrieve TCP/IP map	S1
RTRV-L1ADR: Retrieve level 1 Address Data Repository	P1
RTRV-L1TDR: Retrieve level 1 topology data repository	P1
RTRV-LOG-ALM: Retrieve log alarm	S1
RTRV-LOG-NNEVT: Retrieve NN event log	S1
RTRV-LOG-NTFCN: Retrieve notification log	S1
RTRV-LOG-PROTNSW: Retrieve log protection switch	S1
RTRV-LOG-SECU: Retrieve log security	S5
RTRV-LOPTF: Retrieve low-order path termination function	P1
RTRV-LONIM: Low-order non-intrusive monitor	P1
RTRV-LOXCPM: Retrieve performance monitoring data of LOXC pack	PM1
RTRV-LPBK (cross-connection): Retrieve loopback (cross-connection)	T1
RTRV-LPBK (facility): Retrieve loopback (facility)	T1
RTRV-MANROUTE-MCN: Retrieve manual IP route for management communication	S1
RTRV-MAP-NEIGHBOR: Retrieve map neighbor	S1
RTRV-MAP-NETWORK: Retrieve network map	S1
RTRV-MAP-RING: Retrieve map ring	M1
RTRV-NADR: Retrieve ONNS network (domain) address data repository	P1
RTRV-NE: Retrieve NE	S1
RTRV-NE-SECU: Retrieve NE security	S1
RTRV-NNCNFG: Retrieve ONNS configuration data	P1
RTRV-NNDR: Retrieve ONNS neighbor nodal data repository	P1
RTRV-NNEVT-TCA	P1
RTRV-NTDR: Retrieve ONNS network topology data repository	P1
RTRV-OC3: Retrieve OC3 port	P1
RTRV-OC12: Retrieve OC12 port	P1
RTRV-OC48: Retrieve OC48 port	P1

TL1 command	User privilege code (UCFC/UCAL)
RTRV-OC48T: Retrieve OC48T port	P1
RTRV-OC192: Retrieve OC192 port	P1
RTRV-OTU2: Retrieve OTU2 port	P1
RTRV-OUC: Retrieve ONNS to UPSR/SNCP constructs	P1
RTRV-OW: Retrieve orderwire and user bytes	M1
RTRV-PATH: Retrieve ONNS connection data	P1
RTRV-PM: Retrieve performance monitoring	PM1
RTRV-PREDEFRAG: Perform an ONNS fragmentation analysis	P1
RTRV-PRMTR-DATA: Retrieve parameter data	S1
RTRV-PRMTR-SFTWR: Retrieve parameter software	S1
RTRV-PROTN-GRP: Retrieve protection group	M1
RTRV-PTF: Retrieve tributary with path termination function	P1
RTRV-RC-GENERAL: Retrieve general routing controller settings	S1
RTRV-RC-NEIGHBOR: Retrieve all neighboring routing controllers	S1
RTRV-RESALLOC-VCG: Retrieve resource allocation virtual concatenation group	P1
RTRV-RSVP: Retrieve RSVP parameters	S1
RTRV-SCN-NEIGHBOR: Retrieve SCN neighbor	S1
RTRV-SPANSWS: Retrieve SPAN switch state	M1
RTRV-STATE-EQPT (CP): Retrieve equipment state (circuit pack)	M1
RTRV-STATE-EQPT (slot): Retrieve equipment state (slot)	M1
RTRV-STATE-EQPT (socket): Retrieve equipment state	M1
RTRV-STATE-EQPT (system): Retrieve equipment state (system)	M1
RTRV-STM1E: Retrieve STM1E port	P1
RTRV-STM1: Retrieve STM1 port	P1
RTRV-STM4: Retrieve STM4 port	P1
RTRV-STM16: Retrieve STM16 port	P1
RTRV-STM16T: Retrieve STM16T port	P1
RTRV-STM64: Retrieve STM64 port	P1
RTRV-STS1: Retrieve STS1 tributary	P1
RTRV-SYNCN: Retrieve synchronization	M1
RTRV-T3: Retrieve DS3 port	P1
RTRV-TACC: Retrieve test access	T1
RTRV-TCA-ASGNMT: Retrieve TCA assignment	PM1

TL1 command	User privilege code (UCFC/UCAL)
RTRV-TCA-PROF: Retrieve TCA profile	PM1
RTRV-TMAT3: Retrieve asynchronous adjunct port of the TransMUX card	P1
RTRV-TMST3: Retrieve synchronous adjunct port of the TransMUX card	P1
RTRV-TMUXPM: Retrieve performance monitoring data of TransMUX pack	PM1
RTRV-ULS: Retrieve upper layer stack	S1
RTRV-ULSDCC-L3: Retrieve layer 3 related parameters	S1
RTRV-ULSDCC-L4: Retrieve OSI layer 4 related parameters	S1
RTRV-USER: Retrieve user	S2
RTRV-USER-SECU: Retrieve user security	S4
RTRV-VC3: Retrieve VC3 tributary	P1
RTRV-VCG: Retrieve virtual concatenation group	P1
RTRV-VCG-TRANS: Retrieve virtual concatenation group transparent	P1
RTRV-VCGTRIB: Retrieve virtual concatenation group tributary	P1
RTRV-VRTSW: Retrieve virtual switch	P1
SET-ATTR-CONT: Set attribute control	M3
SET-ATTR-ENV: Set attribute environment	M3
SET-MAP-RING: Set map ring	M3
SET-OW: Set orderwire and user bytes	M3
SET-SID: Set SID	S4
SET-SYNCN: Set synchronization	M3

16 Amendments

Overview

Purpose

The amendments are divided into:

- general amendments
- TL1-related amendments.

The syntax of the amendment number (for example, “Amendment 2.0 -nb12345”) is as follows:

- first part: the amendment is known since this release
- second part: internal number.

Contents

General amendments

Overview

Purpose

This section lists the general amendments.

Contents

General

Amendment 9.0 – nb74169

Traffic hits of about 2 ms when plugging the standby XC

When plugging the standby XC pack unexpected traffic hits of about 2 ms can be observed.

Amendment 10.0 – nb80809

Temporary CPFail alarm for standby CTL on autonomous reboot

When the standby CTL reboots autonomously after it was operational (that means *opstate* = OS_Enabled) a temporary `CPFail` alarm is raised for this pack. This alarm clears again after about two minutes, when the standby CTL has started the initial DB synchronization of the CF-Disk contents (that means *opstate* = OS_SYNCHING).

An autonomous reboot of the CTL may happen, for example, due to the following:

- a program exception on standby CTL
- detection that the CF-Disk synchronization does not work anymore

In case of an operator-driven reboot through TL1, `CPFail` is not raised.

Configuration management

Amendment 3.0.1 – nb19693

Unallowed changes of cross-connection parameters while remote test access sessions are provisioned are not rejected properly

If the operator tries to change the interface standard or tributary mode while a test access session for this cross-connection is active then a slightly wrong error message will be returned.

Amendment 3.0.1 – nb03284

IDNV TL1 error code without parameter name

A simple parameter problem within a TL1 command (for example, parameter out of range) will result in an anonymous IDNV error, without mentioning the affected parameter:

```
IDNV /*Input, Data Not Valid */
```

In most cases the TL1 commands are hidden behind the GUI (of for example, *WaveStar®* CIT or EMS), which does those checks up-front or does not allow to enter wrong values.

Amendment 8.0 – nb56630

Not possible to set lower-order switch capacity and ASAPs for LOXC pack together

If the command **`ED-EQPT`** is executed for a LOXC pack which has the lower-order switch capacity (parameter *loswcap*) and the ASAPs (parameters *eqptpn* and *eqptprotpn*) specified, then the command is denied.

Workaround

The operator should change lower-order switch capacity and ASAPs in two separate commands.

Optical network navigation system (ONNS)

Amendment 9.1 – ch00148

Changing of interface standard for ONNS ports may be denied, although all requested conditions are fulfilled

In some cases changing the interface standard of a port of port class other than TRADITIONAL may be denied, although all requested conditions are fulfilled (for example, no cross-connection existing and port control in LOCKOUT).

In detail the cases are:

- change from SONET to SDH for ports with port class INNI
- change from SDH to SONET for all ports with port class different from TRADITIONAL, in case the system default tributary mode is ADAPTIVE
- change from SDH to SONET for ports with port class INNI, in case the system default tributary mode is FIXED

Workaround

In case such a rejection occurs, the following should be applied to an affected port in the given sequence:

1. set port class of affected port to TRADITIONAL
2. change interface standard of affected port as required
3. set port class of affected port back to original value

Protection (BLSR/S-SPRing, DRI/DNI, APS/MSP, UPSR/SNCP)

Amendment 8.0.1 – nb68076

RTRV-CKT does not return the actual AID for add and drop tributaries in the local node

The Retrieve Circuit TL1 command (**`RTRV-CKT`**) provides the view of the local node on MS-SPRing / BLSR service and extra traffic circuits.

Currently the command does not return the actual AID of tributaries with ring add or drop cross-connections in the local node. Instead the string “REMAID” is returned as it is also done for tributaries of add and drop cross-connections in remote nodes.

Amendment 8.0.1 – nb68388

RTRV-CKT command on non existing tributary does not complete

If RTRV-CKT is used for a tributary, which does not exist (for example, at an empty/unprovisioned slot), then the command does not complete. That means, after the IP (in progress) response there is no further output.

This behavior will lead to time out of the command on management system or *WaveStar®* CIT. The user will not get an indication, why the command failed. The expected response from the system would be an IENE error code.

Area DCN/DCC management

Amendment 3.0.1 – nb17090

No specific error message when 64 DCC channel limit is reached

When the maximum number of available DCC channels is reached, the supplied error message does not distinguish the possible reasons and always replies with:

`SSRE, System Resources Exceeded, allowed limit of DCC channels exceeded.`

There is no distinction if the number of

- SCN (Signaling Communication Network) assignments or
- MCN (Management Communication Network) assignments or
- a combination of both.

is exceeded.

TL1 amendments

Overview

Purpose

This section lists the amendments related to the TL1 commands.

Contents

Administration

Amendment 10.0 – nbnb81953

Missing user ID in forced logout notification

If a user is logged out by a security administrator using the **`CANC-USER-SECU`** command, then the security notification indicating the forced logout does not contain the user ID of the logged out user.

Impact: Other users receiving the security notification do not know which user has been logged out.

Transmission and cross-connection handling

Amendment 9.0 – ch00071

Wrong error code when creating conversion cross-connection with not allowed parameters

If an AU3/AU4 conversion cross-connection is created, and path protection related parameters are specified in the **`ENT-CRS`** command, then the command is denied with one of the wrong error codes:

`Status – Not in Valid State, Parameter(s) present which are not allowed`

`Status – Command Not Found, command not recognized or command parameters not valid`

The expected error response is:

`Input – Data Not Valid, Parameter(s) present which are not allowed`

Amendment 9.1 – nb58910

Running LOXC script causes system to respond with “Status, All Resources Busy, System limit exceed”**, setup or roll of lower-order cross-connections on LOXC40 might fail when issued to fast, e.g. via TL1 script**

When setting up or rolling a lot of lower-order cross-connections being switched on a LambdaUnite NE equipped with LOXC40 it must be avoided to issue the **`ENT-CRS`** or **`ENT-ROLL`** commands too fast.

If the command sequence is sent to the NE too fast some lower-order cross-connection might not be set up correctly on LOXC40 even though all commands are correctly acknowledged by the COMPLD indication.

A reboot by applying a controller reset to the CTL or to the LOXC40 cleans up configuration and ensures to have all cross-connections set up correctly in HW. Waiting for the COMPLD is not sufficient.

When using the ranging support of *WaveStar*® CIT to set up lower-order cross-connections, the issue does not occur because at most 63 lower-order cross-connections can be set up with a single request which does not yet lead to the overload condition on LOXC40.

Workaround:

When issuing a lot of **`ENT-CRS`** commands to set up lower-order cross-connections or **`ENT-ROLL`** commands to roll lower-order cross-connections (for example, through TL1 script),

- limit the number of instances for one command series to 63,
- make the script wait for the NE reaction (COMPLD or DENY) after each single TL1-command sent to the Unite NE before sending the next one, and
- add delays of 500 ms, especially in the case of VC12 and VT1.5 connections, 800 ms in the case of LO-VC3 connections between the e.g. **`ENT-CRS`** commands.

See script examples adding delay below.

Sample of current script:

```
#delay=500
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-1-192-28,1-1-#-#-35-2-1-1:26882::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-1,1-1-#-#-35-2-1-2:26883::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-2,1-1-#-#-35-2-1-3:26884::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-3,1-1-#-#-35-2-1-4:26885::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-4,1-1-#-#-35-2-1-5:26886::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-5,1-1-#-#-35-2-1-6:26887::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-6,1-1-#-#-35-2-1-7:26888::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-7,1-1-#-#-35-2-1-8:26889::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-8,1-1-#-#-35-2-1-9:26890::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-9,1-1-#-#-35-2-1-10:26891::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-10,1-1-#-#-35-2-1-11:26892::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-11,1-1-#-#-35-2-1-12:26893::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-12,1-1-#-#-35-2-1-13:26894::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-13,1-1-#-#-35-2-1-14:26895::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-14,1-1-#-#-35-2-1-15:26896::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-15,1-1-#-#-35-2-1-16:26897::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-16,1-1-#-#-35-2-1-17:26898::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-17,1-1-#-#-35-2-1-18:26899::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-18,1-1-#-#-35-2-1-19:26900::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-19,1-1-#-#-35-2-1-20:26901::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-20,1-1-#-#-35-2-1-21:26902::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-21,1-1-#-#-35-2-1-22:26903::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-22,1-1-#-#-35-2-1-23:26904::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-23,1-1-#-#-35-2-1-24:26905::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-24,1-1-#-#-35-2-1-25:26906::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-25,1-1-#-#-35-2-1-26:26907::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-26,1-1-#-#-35-2-1-27:26908::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-27,1-1-#-#-35-2-1-28:26909::1WAY;
#delay=500
```

A) adding a delay of 500 ms or more if required, will prevent the CTL reboot

Example:

#delay=500, where the delay value is in milliseconds (that is, 500 ms).

B) adding delay more frequently through the script, will also prevent the CTL reboot

Example:

```
#delay=500
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-1-192-28,1-1-#-#-35-2-1-1:26882::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-1,1-1-#-#-35-2-1-2:26883::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-2,1-1-#-#-35-2-1-3:26884::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-3,1-1-#-#-35-2-1-4:26885::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-4,1-1-#-#-35-2-1-5:26886::1WAY;
#delay=500
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-5,1-1-#-#-35-2-1-6:26887::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-6,1-1-#-#-35-2-1-7:26888::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-7,1-1-#-#-35-2-1-8:26889::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-8,1-1-#-#-35-2-1-9:26890::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-9,1-1-#-#-35-2-1-10:26891::1WAY;
#delay=500
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-10,1-1-#-#-35-2-1-11:26892::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-11,1-1-#-#-35-2-1-12:26893::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-12,1-1-#-#-35-2-1-13:26894::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-13,1-1-#-#-35-2-1-14:26895::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-14,1-1-#-#-35-2-1-15:26896::1WAY;
#delay=500
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-15,1-1-#-#-35-2-1-16:26897::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-16,1-1-#-#-35-2-1-17:26898::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-17,1-1-#-#-35-2-1-18:26899::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-18,1-1-#-#-35-2-1-19:26900::1WAY;
ENT-CRS-LOVT1:WESTFORD:1-1-#-#-35-2-1-19,1-1-#-#-35-2-1-20:26901::1WAY;
#delay=500
```

Should the system still exhibit undesired behavior, increase the delay times by a factor of at least 2. If there is a suspicion that not all lower-order cross-connections have been setup as expected, apply a controller reset to the CTL or LOXC pack.

Amendment 9.1 – nb74339

Wrong error code when creating conversion cross-connection with not allowed parameters

If an AU3/AU4 conversion cross-connection is created, and path protection related parameters are specified in the **ENT-CRS** command, then the command is denied with one of the wrong error codes:

```
Status – Not in Valid State, Parameter(s) present which are not allowed
Status – Command Not Found, command not recognized or command
   parameters not valid
```

The correct error code is:

```
Input – Data Not Valid, Parameter(s) present which are not allowed
```

Amendment 10.0 – ch00243

Wrong error when setting outgoing J0 section trace with wrong length

If it is tried to set the outgoing J0 section trace to a value with a length that does not fit to the J0 format specified in the same command, then the **ED-rr** command is denied with the wrong error code:

```
IDNV – Input, Data Not Valid, invalid strcout
```

The correct error code is:

```
IDNC – Input, Data Not Consistent, strcout length must match the given
   outgoing J0 trace mode
```

Amendment 10.0 – nb83216

GE1: Unsupported parameter rcf_htmr reported

When provisioning a GE1 pack, the parameter *rcf_htmr* is reported with the **ENT-EPORT** notification even though it is not supported by this pack. In addition, when retrieving the parameters of a LAN port of a GE1 pack using the command **RTRV-EPORT**, this parameter is erroneously reported.

When trying to modify it, the **ED-EPORT** command is rejected as expected.

Amendment 10.0 – nb83337

ED-VCG returns incorrect error codes for GE10PL1 related VCGs

The system returns the following incorrect error codes on **ED-VCG** for GE10PL1 related VCGs:

1. Try to set *vcat*="DISABLE" for a 1 GbE port on GE10PL1:
   `SDNC – Status, Data Not Consistent, Not allowed on GE10PL1 pack`
   The correct error code is:
   `SNVS – Status, Not in Valid State, Not allowed for 1 GE interfaces`
2. Try to set *istd*="SONET" for a 1 GbE port on GE10PL1:
   `SDNC – Status, Data Not Consistent, Not allowed on GE10PL1`
   The correct error code is:
   `IDNV – Input, Data Not Valid, Interface Standard cannot be modified`

3. Try to set *inputsig*="644" for a 10 GbE port while *vcat* is disabled:
   `SDNC – Status, Data Not Consistent, Not allowed on GE10PL1 pack`
   The correct error code is:
   `SNVS – Status, Not in Valid State, Not allowed if VCAT disabled`
4. Try to set *istd*="SONET" for a 10 GbE port on GE10PL1:
   `IDNV – Input, Data Not Valid`
   The correct error code is:
   `IDNV – Input, Data Not Valid, Interface Standard cannot be modified`

Amendment 10.5 – nb85483

Modification of a GE10PL1-related cross-connection where a specified tributary is not aligned to VC4/STS3C boundary causes a CTL reboot

When modifying a cross-connection using a tributary AID referring to GE10PL1 which is not aligned to a VC4/STS3C boundary (for example, tributary number 192 instead of 190), the CTL performs a reboot, thus losing all management connections. This reboot does not occur with tributary AIDs referring to SDH/SONET ports.

Workaround

Always use aligned tributary AIDs when using commands to modify cross-connections in cut-through mode.

Amendment 11.0 – nb86776

Wrong error code when creating a cross-connection at tributaries which are involved in a test access session

If it is tried to create a lower-order cross-connection with an input or output tributary which is used in a lower-order test access session, then the **`ENT-CRS`** command is denied with the wrong error code:

`SNVS, Status, Not in Valid State, output is already cross connected`

The required error code is:

`SNVS , Status, Not in Valid State, Tributary used in test access session`

Amendment 11.0 – nb86586

Wrong error code when modifying VCG parameters not supported on ES64U

If it is tried to modify parameters of a VGC on an ES64U pack, and the parameters specified in the command are not supported by the ES64U (that means, the parameters can be retrieved but not modified), then the **`ED-VCG-ES64U`** command is denied with the wrong error code:

`SDNC – Status, Data Not Consistent, Not allowed on GE10PL1 pack`

The correct error code would be:

```
SDNC – Status, Data Not Consistent, Not allowed on ES64U pack
```

Amendment 11.0 – nb87553

Wrong error code when creating cross-connections using lower-order tributaries which are used in a lower-order test access session

When trying to create lower-order cross-connections using lower-order tributaries which are used as E-trib or F-trib of a lower-order remote test access session, this request is rejected as required. However, the returned error code is misleading.

```
Status, Not in Valid State, Too many crossconnection originating  from
same tributary
```

The correct error code would be:

```
Status, Not in Valid State, tributary used in test access session
```

Amendment 11.0 – nb87448

DBCHG ENT-CRS-LOxx missing parameters for converting 1way to 2way for lower-order rates

The database change notification of the TL1 command **`ENT-CRS-LO`*`xx`*** for the creation of lower-order cross-connections does not follow the specification. The NE does not report the parameters *rtnxcappl*, *rtnxcnum*, and *rtnomode* in the **`DBCHG`** notification, if an existing 1way lower-order cross-connection is converted to a 2way lower-order cross-connection by the creation of a appropriate cross-connection in the opposite direction.

Workaround

Use the **`RTRV-CRS-LO`*`xx`*** command to retrieve the parameters.

Configuration management

Amendment 9.1 – ch00133

Parameters missing in RTRV response and ENT notification for OTN client port

The problem is related to the OC192/STM64 OTN client port on an OP10/XTT pack. In the ENT-OC192/ENT-STM64 notification for an OTN client port the ENNI parameters *clientNodeID*, *linkResCls*, *localLinkID*, *localLPID*, *neighborSigPCID*, *neighborSigPCIPAddr*, *remAdvNodeID*, *remLinkID* and *remRAID* are missing.

Because the OP10/XTT pack will not be used with ENNI in R9.1 and later, this problem is acceptable for customers. In the RTRC-OC192/RTRV-STM64 response for an OTN client port the parameters *opdis* and *opwl* are missing. The parameters *opdis* and *opwl* are always set to "NA" for client ports at the OP10/XTT, the customer will not loose relevant information.

Amendment 9.1 – nb77810

Wrong error code when path protected cross-connection is converted to 1-Way without specifying the leg to be dropped

When **CNVT-CCT** command is used to convert a lower-order path protected cross-connection to a 1-way cross-connection, and the parameter *in_aid2* with the leg to be dropped is not specified in the command, then the command is denied with the wrong error code:

```
SNVS – Status, Not in Valid State, Parameter(s) present which are not
   allowed
```

The preferred error code for a missing parameter is:

```
IPMS – Input, Parameter Missing, <parameter name> is missing
```

Because *in_aid2* is one of the command parameters, also this error code would be possible:

```
SCNF – Status, Command Not Found, command not recognized or command
   parameters not valid
```

Amendment 10.0 – nb82887

Wrong error code if CNVT-CCT is used with incorrect spec block parameter

When a **CNVT-CCT** command is applied to convert an unprotected cross-connection into a path-protected one using an incorrect value for the SNC type parameter, then the command is denied with the wrong error code:

```
SCNF /* Status, Command Not Found, command not recognized or command
   parameters not valid */
```

The required error code is:

```
IDNV /* Input, Data Not Valid */
```

Amendment 10.0 – nb83091

Unsupported parameters reported with RTRV-PTF-HOSTS1 and the ENT-PTF-HOSTS1 notification

For an STS1 PTF, which has been created by lower-order substructuring (modifier HOSTS1), some parameters are reported with **`RTRV-PTF-HOSTS1`** and in the **`ENT-PTF-HOSTS1`** notification which are not supported and should therefore be omitted. The wrongly reported parameters are:

`ENT-PTF-HOSTS1` notification: *dexcthr*, *feppm*, *neppm*, *pmhpfeses*, *pmhpneses*

`RTRV-PTF-HOSTS1`: *feppm*, *neppm*, *pmhpfeses*, *pmhpneses*

Amendment 10.0 – nb83149

Wrong error code if port of a 2F MS-SPRing/BLSR protection group is set to ENNI

If it is tried to set the port class of a port in a MS-SPRing/BLSR protection group to ENNI, then the **`ED-rr`** command is denied with the wrong error code:

```
SNVS – Status, Not in Valid State, port used for MSSPRING/BLSR
```

The required error code is:

```
SNVS – Status, Not in Valid State, Port must not be a member of a BLSR/
   MS-SPRing protection group
```

Amendment 11.0 – nb84678

Wrong error code on ED-LOPTF with invalid explptrc

If the **`ED-LOPTF-TMSVT`** command is used with an invalid value specified for the parameter *explptrc*, then the command is denied with the wrong error code:

```
IDNV – Input, Data Not Valid, invalid exphptrc
```

The parameter name in the error response is not correct.

The required and preferred error code is:

```
IDNV – Input, Data Not Valid, invalid explptrc
```

Optical network navigation system (ONNS)

Amendment 5.0.1 – nb31503

Grouped NN connection setup via TL1 cut-through accepted with egress EDGE port not being a VCG

When using TL1 cut-through commands it is possible to switch a group of more than one NN connection with ingress EDGE port being a VCG as required, but with the EDGE port being a standard SDH/SONET port.

The connections will be switched according to the ingress and egress tributary number association restrictions as defined for grouped connection setup between VCGs, i.e. there will be a one-to-one mapping of the egress tributary numbers to the ingress tributary numbers selected, egress tributary numbers will not be selectable separately.

Thus, connections out of this grouped setup might not be switched, if for example, corresponding egress tributaries are occupied already. Connections that are switched, however, will work properly.

Workaround

Be careful in choosing egress EDGE VCG identifier, if using **`ENT-PATH`** command in cut-through mode to execute grouped NN connection setup with VCGs.

Connections that might have been set up by accident between a VCG and an SDH/SONET EDGE port can be deleted using the normal connection deletion facility.

Amendment 7.0 – nb42305

Wrong notifications for autorv-connections

Some information in the PTHEVT notifications of a restored autorevertive connection by a signal change from no incoming signal to a valid incoming signal is misleading.

The event text of the PTHEVT notification of the failed path of a restored autorevertive connection is incorrect and shall be ignored. The text should indicate that the failed path of a successful restored connection is still available.

Amendment 8.0 – nb51731

Misleading error-message in case of failed path-roll

When setup a path with 3 legs using ED-PATH the following error message can occur:

```
Error=E018:\" Status, Edge Port Used up \":NodeID=29.54.173.73&1-1-5-1-4
```

Amendment 8.0 – nb61009

Inconsistent PTHEVT if trib is outside portrange

In case an ONNS connection setup is issued with an EDGE tributary on a GE1 port given, that is out of the allowed range, additionally to the expected rejection with the failure reason “Invalid EDGE port” and a corresponding **REPT PTHEVT** notification with the same failure reason a second **REPT PTHEVT** notification is sent with the incorrect and misleading failure reason “EDGE port used up” is sent.

Amendment 9.1 – nb73792

RTRV-CONN command completed instead of denied

A **RTRV-CONN** command with an invalid connection ID or an invalid TNA-address or a not matching connection ID and TNA-address is completed without output instead of denied.

Amendment 9.1 – nb78136

SRLGS not shown in output of RTRV-L1TDR

The TL1 command **RTRV-L1TDR** does not show shared risk link groups (SLRGs).

Amendment 9.1 – nb78498

Link defragmentation (OPR-DEFRAG) executed to ports of port class other than INNI formally accepted or denied with misleading error code

Although link defragmentation including “empty port” is only supported for INNI links, that is, can only be executed to ports of port class INNI, the following misbehavior may be observed:

If **OPR-DEFRAG** executed to a port of port class other than INNI, the action may be formally answered with completion code (thus pretending acceptance) or may be rejected with the misleading error response:

```
E621, No Target Port available
```

However, also if command is wrongly formally accepted, no action will be really executed and no traffic will be affected.

Workaround

Take care to execute link defragmentation (including “empty port”) only to INNI ports. Otherwise ignore responses as described above.

Amendment 9.1 – nb78846

Wrong error code when setting portClass to UNI without necessary parameters specified

If the port class is set to UNI and the parameters *clientNodeID*, *networkTNAAddress*, and *localLPID* are not specified in the command, then the **ED-RR** command is denied with the wrong error code:

`IDNV – Input, Data Not Valid, additional parameters required`

The expected error code is:

```
IDNV – Input, Data Not Valid, UNI parameters must be different to their
   DEFAULT values
```

Amendment 10.0 – nb82588

Missing validations related to allowed values and uniqueness when configuring ONNS SCN IP addresses

When configuring ONNS SCN IP addresses via **`ENT-ULS`**, there are no validations applied in the NE or in the management systems to deny invalid values or value ranges (for example, values 0.0.0.0 or 1.1.1.1) or to check uniqueness of ONNS SCN IP address within a domain. As a consequence, corresponding wrong settings would be accepted.

Such wrongly configured ONNS SCN IP addresses could lead to unpredictable behavior of the ONNS network.

Workaround

Be careful when configuring ONNS SCN IP addresses.

Amendment 10.0 – nb83005

Wrong error code if worker port of a MSP protection group is set to INNI

If it is tried to set the port class of the worker port of a 1:1 MSP to INNI, then the **`ED-rr`** command is denied with the wrong error code:

`SNVS – Status, Not in Valid State, target Port Class must be TRADITIONAL`

The required error code is:

```
SNVS – Status, Not in Valid State, Port must not be member of a
   protection group
```

Amendment 10.0 – nb83318

Unspecific error codes in case of failed network connection setup

In the following cases the unspecific error code `E009 – Service affecting defect` is notified in case of a failure of a network connection setup:

- The egress port specified for the destination domain is not available.
  The correct error code is: `E008 – Invalid Egress address`
- An E-NNI link determined by routing is not available.
  The correct error code is:
  `E006 – Selected port on eNNI link already in use`

Amendment 10.0 – nb83357

Failure notifications may be unspecific during network connection setup failures

Notifications for **`ENT-CONN`** do not correctly distinguish on which port (EDGE or E-NNI) the problems are. This is especially the case if the domain only consists of one node.

In case a network connection setup fails due to a problem in a domain with coinciding A-node and Z-node for this connection, corresponding error notifications may be unspecific in mixing up `E006 – Selected port on eNNI link already in use` and `E007 – Invalid egress address`.

Amendment 10.5 – nb85399

Value "N/A" not supported for parameter ″protType″ in ENT-CONN

Within the **`ENT-CONN`** command, the value "N/A" for parameter *protType* is not supported. It is only possible to request either purely unprotected (value 0x1) or 1+1 protected (value 1+1) ENNI links for the requested network connection. It is not possible to specify that all available ENNI links irrelevant of their protection type can be used.

Amendment 10.5 – nb85192

RCDEAD alarm contains neighbor RC IP address instead of RA ID

The RCDEAD alarm raised by a routing controller in case of a routing controller (RC) in a neighbor routing area (RA) not being reachable will contain the IP address of the affected neighbor RC instead of the corresponding RA ID.

Workaround

As long as RC redundancy is not implemented, there is a 1 to 1 relationship between RA ID and RC IP address.

Amendment 10.5 – nb85411

Inconsistent or corrupt reporting of NO-IN condition clearing for ONNS connections

For the various types of ONNS connections, reporting of connection state transition from CON&NO-IN to CON&OK is inconsistent: For types AUTONR, N1PLUS1, and P1PLUS1, this transition will be reported as "Connection is OK", while for other types it will be reported as "EDGE port, IN Signal Fault has cleared". For type YCONN, the corresponding notification may even be corrupted.

Amendment 10.5 – nb85593

Misleading error notification in case of network connection setup failure due to unknown TNA

In case the setup of a network connection is denied due to a unknown TNA specified, the setup will be denied correctly. However, a **REPT CONNCHG** notification may be received with the misleading error code:

`E009, Service affecting defect – No route to destination found`

Amendment 11.0 – nb86788

Misleading error code in case of failed ONNS connection setup with certain constraints

In case of failure of an ONNS connection setup with given exclude and/or explicit constraints due to insufficient bandwidth, this failure may sometimes be reported with the misleading error response

`E033, INLI, Node list has Invalid/unknown ID`

instead of

`E001, ISNR, Insufficient Network Resources for conenction setup`

This misbehavior occurs if the given setup constraints are such that due to these constraints an explicit node given or the Z-node is not rereachablet all.

Protection (BLSR/S-SPRing, DRI/DNI, APS/MSP, UPSR/SNCP)

Amendment 3.0.1 – nb 18762

1+1 MSP accept incomplete Operate Protection Switch commands

For 1+1 MSP, incomplete Operate Protection Switch commands are accepted:

- Operate Protection Switch command with parameter source *sc=lockout* and without parameter destination *dest* is accepted. This is not a real problem, because the *dest* is redundant information.
- Operate Protection Switch command with parameter destination *dest=wkg* should not be accepted without the source information. The resultant state Manual Switch is not worse because it does not prevent further automatic switching.

Amendment 6.0 – nb35660

Wrong error code if tributary of OPT2G5 is used as input for path protection

If a path protected cross-connection is created with a tributary of an OPT2G5 as protection leg, then the command is denied with the unspecific error code:

```
IDNV – Input, Data Not Valid
```

The expected error code is:

```
IDNC – Input, Data Not Consistent, no OPT2G5-trib as input-trib in
   pathprotected crossconnection
```

(This correct error code is returned if the working leg is on an OPT2G5.)

Amendment 6.0 – nb40929

Unprecise error message for ED-MAP-RING

When the TL1 command **`ED-MAP-RING`** is used to provision the MS-SPRing / BLSR ring topology and different numbers of ring node identifiers and ring node target identifiers are entered, the command is denied. However, the related error message is not precise enough, but it simply states:

```
Input, Data Not Valid
```

The expected error code is:

```
Input, Data Not Valid, different number of values in rnids and rtids
```

Amendment 8.0 – nb64251

Wrong error code when locz/loca is entered on creation of a higher-order cross-connection to/from the protection bandwidth of an MS-SPRing / BLSR

When the *locz* parameter is present for the **ENT-CRS** command which tries to provision a higher-order add cross-connection to the protection bandwidth of a 2 fibre MS-SPRing / BLSR or when the *loca* parameter is present for an attempt to provision a higher-order drop cross-connection from the protection bandwidth of a 2 fibre MS-SPRing / BLSR, the system denies the command with an incorrect error code:

`IDNV: Input Data Not Valid`

The expected error code is:

```
SNVS – Command refers to protection tributaries of a BLSR/MS-SPRing
   protection
```

Amendment 9.0 – nb74289

Wrong error code when creating TOPEX protection group with SONET ports

When it is tried to create a 4F TOPEX protection group using 4 SONET ports, then the **ENT-PROTN-GRP** command is denied with the wrong error code:

```
SNVS – Status, Not in Valid State, port rate not supported for this
protection type
```

The correct error code is:

```
SNVS – Status, Not in Valid State, SONET port not allowed for TOP
```

Amendment 10.0 – nb83091

Unsupported parameters reported with RTRV-PTF-HOSTS1 and the ENT-PTF-HOSTS1 notification

When it is tried to create a 4F TOPEX protection group using 4 SONET ports, then the **ENT-PROTN-GRP** command is denied with the wrong error code:

```
SNVS – Status, Not in Valid State, port rate not supported for this
protection type
```

The correct error code is:

```
SNVS – Status, Not in Valid State, SONET port not allowed for TOP
```

Fault/Alarm management

Amendment 3.0.1 – nb20283

Changes of tributary output modes for test access sessions are not notified

Changing test modes of a remote test access session does not generate change notification for *eomode* or *fomode*, but the value really changes. However the retrieval of this test access session delivers the correct values.

Amendment 11.0 – nb86806

RTRV-LOG-ALM with reply type DBSYNC does not return log entries

If the **`RTRV-LOG-ALM`** command is used with reply type DBSYNC and no start date and time specified, then the command does not complete (time-out on management systems or *WaveStar*® CIT).

Remote test access

Amendment 3.0.1 – nb19569

Establishment of remote test access for bridged cross-connections is not rejected properly

If an operator tries to create a remote test access session on a cross-connection which is bridged (broadcast) then the operator gets a wrong error message.

Amendment 4.0 – nb20634

Changes of tributary output modes of test access sessions are not notified

Changing test modes of remote test access session does not generate change notification for *eomode* or *fomode* if the value really changes. However the retrieval of this test access session is correct.

Amendment 11.0 – nb87012

Wrong error code when creating lower-order test access in MONEF mode without secondary test access tributary

If the **`CONN-TACC-LOVT1`** command is executed which has the test access mode (*tstmde*) MONEF specified, but no secondary test access tributary (*tatrib2*), then the command is denied with the wrong error code:

```
IENE – Input, Entity Not Exists, Specified tributary/tributaries do not exist
```

DCN/DCC management

Amendment 3.0.1 – nb21562

No specific error message when 64 DCC channel limit reached

When the maximum number of available DCC channels is reached, the supplied error message does not distinguish the possible reasons and always replies with:

```
SSRE, System Resources Exceeded, allowed limit of DCC channels exceeded
```

In case the maximum number is exceeded due to SCN assignments, expected message is:

```
SSRE, System Resources Exceeded, allowed limit of SCN DCC channels
exceeded
```

In case the maximum number is exceeded due to combined MCN+SCN assignments, expected message is:

```
SSRE, System Resources Exceeded, overall limit of DCC channels exceeded
```

Index