Policy Controller Performance Management

What’s new in (I)SN09

Virtual Private Network (VPN) functionality is supported by the Policy Controller for (I)SN09.

Performance management overview

The Policy Controller records operational measurements (OMs) for performance-related data items, formats the data, and supports output to various devices. OMs help determine preventive and corrective maintenance actions, and identify provisioning problems or capacity limitations.

The Policy Controller application has its own set of performance measurements, but uses the OM subsystem from the Nortel General Session Server (NGSS) platform for all OM functions. Use the **omshow** utility to view OM data from the NGSS platform.

OM data recorded on one Policy Controller unit is independent of data recorded on its mate unit. There is no OM data transfer from one unit to another unit during synchronization activities.

Service monitoring

When OMs are associated with Alarms, they can show service level degradation for the Policy Controller, which indicates resources are running low. This information helps determine the corrective action, including equipment repair.

Operational measurements for Policy Controller

This section describes operational measurement for the Policy Controller.

OM Groups and Registers

A register is a peg counter in the Policy Controller software that stores counts for events. Each register has a name that contains a maximum of 32 alphanumeric characters. Up to 32 registers are grouped logically into an OM group.

An OM group is a logical collection that contains a maximum of 32 related OM registers. Each OM register can only be associated with one OM group, but peg counts for OM groups can be sorted into more than one class. OM groups can belong to more than one class. Some peg counts for OM groups can be queried based on tuples.

The Policy Controller application defines the OM groups and the fields. The user cannot define OM groups. The Policy Controller has two types of OM groups:

1. **Static OMs** – static performance measurements create the counter once the Policy Controller application starts and exist while the Policy Controller application operates. The following Policy Controller OM groups are static OMs:
    - SPC_CALLP on page 15 – records call processing events (half call attempts, successful half calls)
    - SPC_RSRV_REQ on page 27 – records reservation request events (incoming, success, failure, error)
    - SPC_CMMT_REQ on page 23 – records commit request events (incoming, succss, failure, error)
    - SPC_DEL_REQ on page 31 – records deletion events (incoming, success, error)
    - SPC_TOPO_UPDATE on page 17 – records Topology Manager update events from Web and OSS provisioning (incoming, failure)
2. **Per-Instance OMs** – dynamic performance measurements captured for objects such as network nodes. The following Policy Controller OM group is the only per-instance OM group:
    - CAC_PER_NETWORK_ZONE on page 21 – records Call Admission Control (CAC) events to each Network zone (sent, success, failure)

OM classes

There are two OM classes: ACTIVE and HOLDING. The system collects OM data during a specified time interval (called a *holding period*) and stores the data in the active registers. During the collection interval, software processes that run within the Policy Controller application increase or peg the registers in the ACTIVE class for all OM groups. When the collection interval has elapsed, a system process moves the collected data of the active registers into the holding registers for all OM groups.

Also, the class variable LASTFIVE is defined for holding only the last five minutes of active register peg counts for all OM groups. When querying OM groups by class, the user can query the last five minutes of peg count activity for the selected group using LASTFIVE.

The *transfer or holding periods* (synchronized with the collection period) used on Policy Controller are 15 or 30 minutes, with 15 minutes being the default. The alignment of the holding period is always to the top of the hour and the OM reporting intervals are multiples of the holding period.

At the end of each transfer/collection period, the counts in Active class registers transfer to Holding class registers. The software processes clear (zero) the Active class registers to reset them, so they can begin counting events in the next collection period. The Active class registers always contain counts for the current holding period. The Holding class registers always contain counts from the previous holding period. Data in both the active and holding registers is available for review using the **omshow** command.

As an example, if the holding period time is 30 minutes and the current time is 4:50 PM. The active registers represent 4:30-4:50 PM and the holding registers represent 4:00-4:30 PM. Finally at 5:00PM the active registers are copied to the holding registers.

In addition to the above example, the values of OM groups and their registers can be output for the current five minute interval. For example, if the current time is 4:04:39 PM, the register values for the last 4 minutes and 39 seconds can be output using the LASTFIVE option.

Register precision

Registers in the ACTIVE and HOLDING classes are 32-bit, double precision registers. Currently, the Policy Controller does not use 16-bit, single precision registers. 32-bit, double precision registers have a peg count capacity of 4,294,967,296 counts. 16-bit, single precision registers have a peg count capacity of 65,535 counts.

Tuple OM group

OM groups can have multiple collections of register counts for each register class (ACTIVE, HOLDING, LASTFIVE). These collections are referred to as tuples. The peg counts of each tuple are associated with a logical entity.

OM tuple key name and key number

A key name and key number are associated with each tuple defined for an OM group. On the Policy Controller, static OMs have only one tuple, the tuple key is 0 and the tuple name is empty.

Limitations of recording and displaying operational measurements

OM registers pegged for an application such as the Policy Controller application may not reach the OM subsystem for a full minute. This impacts what is displayed when using the **omshow** command for static OM groups.

OM group register data pegged on one unit of a Policy Controller is independent of register data on its mate unit. Register data is not transferred from one unit to another.

Initial configuration of performance and operational measurements

By default, the collection/holding period is set to 15 minutes but can be changed to 30 minutes. There are no other OM parameters that can be modified.

Accessing Policy Controller OMs

Static OMs

You can only view Policy Controller static OMs from the IEMS manager or from an SSH (secure shell) connection to the active or standby Policy Controller units.

To view static OMs, access the OM viewing utility **omshow** from the Policy Controller CLI (command line interface). Also, view the comma separated value (CSV) history files with the CLI using standard unix cat or vi commands.

For more information about using Integrated EMS to access CLI, refer to procedure *Access Policy Controller/NCGL GUIs or CLI using the Integrated EMS* in NTP Policy Controller Security and Administration, NN10434-611.

For more information about accessing OMs using SSH, refer to procedure *Remote login to Policy Controller using a secure shell (SSH)* in NTP Policy Controller Security and Administration, NN10434-611.

Per-Instance OM

There is only one Policy Controller per-instance OM group: CAC_PER_NETWORK_ZONE

Although the NGSS OM subsystem supports dynamic OM groups, the maximum size it supports is not large enough for the OM group

CAC_PER_NETWORK_ZONE. Therefore, the per-instance OM group does not support the **omshow** command.

You can only view the Policy Controller per-instance OM from the Policy Controller Web GUI using the following path:

`Session Server->Monitoring->Session Policy Controller->SPC OM log`

The Policy Controller per-instance OM data is stored in folder:

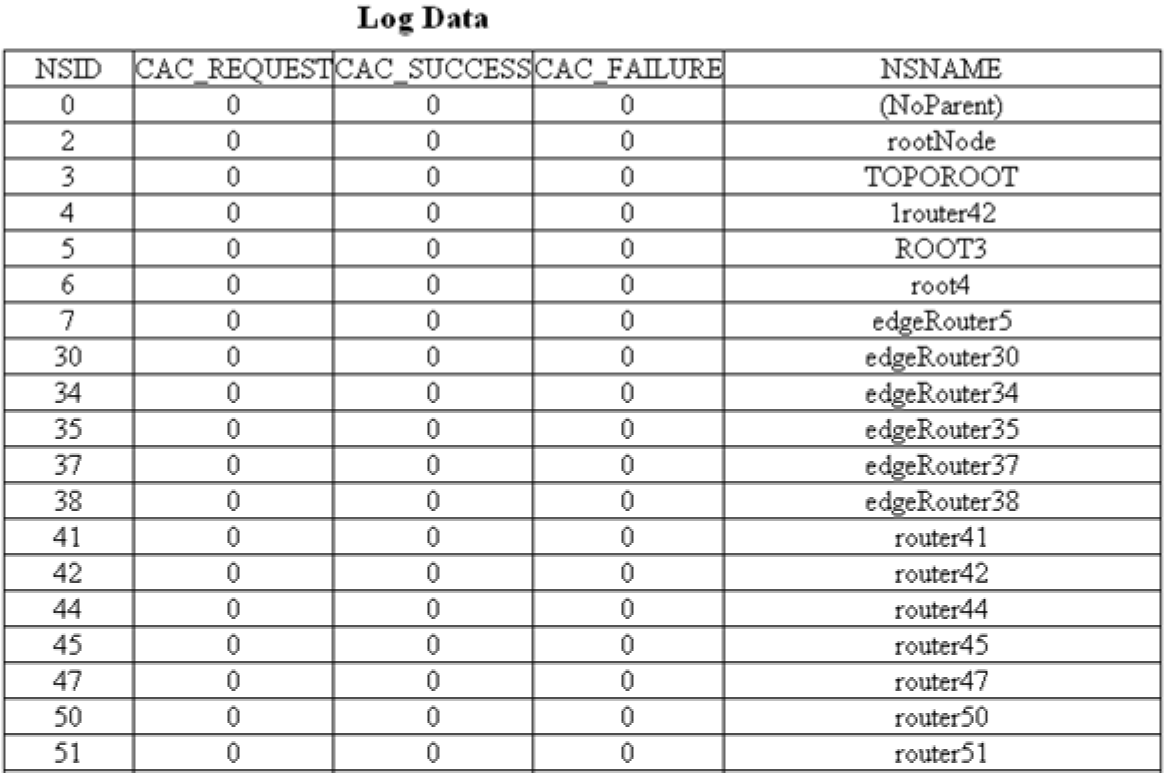
`/opt/apps/ngsspm/nsom/`

File name format: **`SpcNsOm_<date>.dat`**
Example: **`SpcNsOm_2005.04.27.dat`**

The Policy Controller per-instance OM data file generates every 24 hours. The **SpcNsOM_2005.04.27.dat** file, records OM summary data for the date 2005.04.27. A maximum of 10 files will be kept on the Policy Controller unit. Per-instance OM data doesn't sync between the Policy Controller units. The file format is text and you can view its content from the Policy Controller Web GUI through the SPC OM log link.

The following figure is an example of the SPC OM log (accessed from the Web GUI) showing the per-instance OM CAC_PER_NETWORK_ZONE

SPC OM Log output



Log Data

NSID	CAC_REQUEST	CAC_SUCCESS	CAC_FAILURE	NSNAME
0	0	0	0	(NoParent)
2	0	0	0	rootNode
3	0	0	0	TOPOROOT
4	0	0	0	1router42
5	0	0	0	ROOT3
6	0	0	0	root4
7	0	0	0	edgeRouter5
30	0	0	0	edgeRouter30
34	0	0	0	edgeRouter34
35	0	0	0	edgeRouter35
37	0	0	0	edgeRouter37
38	0	0	0	edgeRouter38
41	0	0	0	router41
42	0	0	0	router42
44	0	0	0	router44
45	0	0	0	router45
47	0	0	0	router47
50	0	0	0	router50
51	0	0	0	router51

For information on using the Policy Controller GUI, refer to NTP Policy Controller Configuration Management, NN10432-511.

OM information output to CSV files

The **omshow** function provides near term performance information, but not information older than 30 minutes or 60 minutes, depending on the holding period. In certain circumstances, this historical performance information can be valuable. Therefore, OM information is also saved from the holding registers to comma separated value (CSV) ASCII files on the Policy Controller hard drives. Two sets of file groups are created for storing OM information: traffic history files and standard history files.

- **Traffic history files** contain the last 30 minutes of generated OM information. Each of the six traffic files consists of a particular five minute interval within the 30 minute time frame. Locate traffic history files at the following path:

`/opt/apps/ngsspm/trafhist.`

Traffic history files are named using the following format:

`NGSS.TRAF_OMs.QoS.<year>.<month>.<day>.<hour>_<minute><seconds>_<timezone>.csv`

- **Standard history files** contain the last 24 hours of generated OM information. If the OM collection holding period is 15 minutes, then a total of 96 files are kept, representing the last 24 hours of generated OM information. However, if the collection interval is 30 minutes, then only 48 files are kept. Locate Standard history files at the following path:

  `/opt/apps/ngsspm/stdhist.`

  Standard history files are named using the following format:

  `NGSS.STD_OMs.QoS.<year>.<month>.<day>.<hour>_<minute><seconds>_<timezone>.csv`

Sample CSV file contents

```
PMFile=Begin
MeasurementCategoryFileCreationTimeEarliestStartTimeLatestCaptureTime
RegisterTypeDefaultValueTypeDefault
pm       2005.2.22_14.15_EDTunknownunknowncounterinteger
System=Begin
SystemId
SessionServer
Entity=Begin
EntityId
47.142.97.115
SubEntity=Begin
SubEntityId
Unit0
Table=Begin
TableId  MeasurementKindIntervalDurationCaptureTimeRealiability
SIPGW_CALLPPeriodBased15unknownValid
Labels=Begin
TupleKey KeyName
Key      KeyName
Reg1Name Reg2NameReg3NameReg4Name
IC_CALL_ATTEMPTSOG_CALL_ATTEMPTSCALLS_ABANDONEDCALLS_ANSWERED
Reg5Name Reg6NameReg7Name
CALLS_REJECTEDCALLS_REDIRECTEDOVRLD_CALLS_REJECTED
Labels=End
RowOfValues=Begin
TupleKey KeyName
0        SIPLINKNAME1
Reg1ValueReg2ValueReg3ValueReg4ValueReg5ValueReg6ValueReg7Value
0        0       00000
TupleKey KeyName
1        SIPLINKNAME2
Reg1ValueReg2ValueReg3ValueReg4ValueReg5ValueReg6ValueReg7Value
0        0       00000
TupleKey KeyName
2049     SIPLINK_UNKNOWN
Reg1ValueReg2ValueReg3ValueReg4ValueReg5ValueReg6ValueReg7Value
0        0       00000
RowOfValues=End
Table=End
         .
         .
         .
         .
GroupVals=End
SubEntity=End
Entity=End
System=End
PMFile=End
```

Policy Controller CALLP OMs generated on the core

There are OMs on the Core used for Policy Controller CALLP measurements. Core OM group, NGSSOM is used for tracking CS2000-to-application server calls (where Policy Controller is the application server) and CS 2000-to-CS 2000 (VRDN) calls.

OM registers in the NGSSOM group are used for tracking the effects of the CS2B0009 SOC limit setting on CS2K-AS (CS2000-to-call server) calls. These are:

- register CS2ASOVF - pegged when a CS2AS Policy Controller call is deflected because allowing the call would have caused the current CS2AS call count to exceed the CS2B009 SOC limit.
- register CS2ASHWM - keeps track of the maximum value reached for the CS2AS call counter.

Other registers used in Core OM group NGSSOM, related to the use of SOC CS2B0008, are used to keep track of CS2CS or Call server-to-call server (VRDN) calls. These are:

- CS2CSOVF - pegged when a CS2CS Policy Controller call is deflected because allowing the call would have caused the current CS2CS call count to exceed the CS2B0008 SOC limit.
- CS2CSHWM - keeps track of the maximum value reached for the CS2CS call counter.

To view CALLP OMs generated on the core, refer to the CS 2000 Performance Management NTP applicable to your solution.

Monitoring the Policy Controller resources for optimal performance

Currently there are no procedures or practices for altering the performance of the Policy Controller or its applications.

Policy Controller platform resources such as memory, CPU and disk drive file system usage use preset thresholds for generating alarms. You can monitor these parameters using procedures in NTP Policy Controller Fault Management, NN10438-911. You can change performance monitoring thresholds for file systems only using NTP Policy Controller Configuration Management, NN10432-511.

Generating performance reports

You cannot generate performance reports for the Policy Controller.

Procedures for viewing operational measurements

The following table lists the available performance management procedures.

Policy Controller performance management procedures

Procedure
View Policy Controller operational measurements on page 11 To view the status of platform resources such as memory and disk usage, refer to procedure *View the operational status of a Policy Controller NCGL platform* found in NTP Policy Controller Security and Administration, NN10434-611. To change the parameters for file system monitoring, refer to NTP Policy Controller Configuration Management, NN10432-511.

View Policy Controller operational measurements

Purpose of this procedure

Use this procedure to display operational measurement (OM) information for the following OM groups:

- SPC_CALLP
- SPC_RSRV_REQ
- SPC_CMMT_REQ
- SPC_DEL_REQ
- SPC_TOPO_UPDATE

Limitations and restrictions

The following restrictions and limitations apply to this procedure:

- You cannot use the **omshow** command to access the per-instance OM CAC_PER_NETWORK_ZONE.
- This procedure cannot be used to access long-term OM information or information that is older than 24 hours.
- Beginning in SN08, OM group parameter ALL is no longer a supported when using the omshow command.

Prerequisites

OMs cannot be viewed directly by the Integrate EMS. To view OMs through the Integrated EMS for element management activities, refer to procedure *Access Policy Controller/NCGL GUIs or CLI using the Integrated EMS*, found in the Policy Controller Security and Administration NTP, NN10434-611, to access a command line interface (CLI).

Refer to procedure *Remote login to Policy Controller using a secure shell (SSH)*, also found in the Policy Controller Security and Administration NTP, NN10434-611, to access a command line interface (CLI) using an available client workstation.

Refer to procedure Using the omshow command on page 12 for details about using the **omshow** command to display all or parts of the OM groups and to change settings related to OM registers and OM holding periods.

Action

Using the omshow command

At a Policy Controller CLI

1 Log in to the Policy Controller as *mtc* user.

2 To view registers for a specific OM group, at the prompt type:

$ **omshow <omgroup> <class> <tuple_info>**

and press Enter.

where

omgroup
a required value, is one of the following OM groups

- SPC_CALLP
- SPC_RSRV_REQ
- SPC_CMMT_REQ
- SPC_DEL_REQ
- SPC_TOPO_UPDATE

class
a required value, is a class of OM group:

- holding (for the holding registers)
- active (for the active registers)
- lastfive (for the last five minutes of active register peg counts)
- zero (zeros out both the active and lastfive registers)

tuple_info
an optional value, is a tuple information parameter:

- tuple key number: enter a tuple key number of 0.
- tuple key range: enter a range of tuple values from 0 to 4,294,967,295, separated by a space. For instance, 0 10 will display tuples 0 through 10.
- tuple key name: leave this blank.
- tuple key name range: Not applicable. Leave this blank

3 Refer to section Additional information - alternate syntax for omshow command on page 13 to review example of other command details.

4 You have completed this procedure.

Additional information - alternate syntax for omshow command

Use the commands in the following table to view different kinds of OM data view tuple and link peg counts, to zero out registers and to set the holding period value.

omshow command variations

omshow command syntax	Description
omshow omshow help	Displays a list of the required command syntax along with a complete list of available OM groups. Several simple command syntax examples are also shown.
omshow <omgroup> active	Displays the “active” class register peg counts for all tuples in the selected OM group.
omshow <omgroup> holding	Displays the “holding” class register peg counts for all tuples in the selected OM group.
omshow <omgroup> lastfive	Displays the last 5 minutes of active register peg counts for all tuples in the selected OM group.
omshow <omgroup> zero	Zeros the active and lastfive registers for a selected OM group.
omshow holdingperiod 30	Sets the holding period to 30 minutes.
omshow holdingperiod 15	Sets the holding period to 15 minutes (default).
omshow <omgroup> lastfive 0	Displays all the register counts from the last five minutes for tuple number 0 in the selected OM group.
omshow <omgroup> active LN2RTPF_IT1	Displays all the active register counts for the tuple key name ln2rtpf_it1, for the selected OM group.
omshow <omgroup> active 0 20	Displays all the active register counts for the range of tuple numbers 0 through 20, for the selected OM group.
omshow <omgroup> active LN_NGSS_TEST LNIT_LOOP1	Displays all the active register counts for the range of tuple key names from LN_NGSS_TEST through LNIT_LOOP1, for the selected OM group. Note: The range order of the tuple key names can be obtained by first using the tuple number option to view a range of tuple numbers and associated tuple names.

SPC_CALLP

Description

Policy Controller Call Processing (SPC_CALLP)

OM group SPC_CALLP is a static OM. It provides registers for recording call processing events for the Policy Controller application.

The following table lists the key and info fields associated with OM group SPC_CALLP.

Key field	Info field
None	None

Related functional groups

Policy Controller static OM groups have only one tuple. The tuple key is 0 and the tuple name is empty.

Registers

The following table lists the registers associated with OM group SPC_CALLP and what they measure.

Registers for OM group SPC_CALLP

Register name	Measures
NUM_HalfCall_Attempts	Total number of incoming half call attempts received by the Policy Controller
NUM_HalfCall_Success	Total number of successful half calls set up by the Policy Controller

NUM_HalfCall_Attempts

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of incoming half call attempts received by the Policy Controller.

Associated registers
IC_RESERVATION_REQUESTS
RESERVATION_REQUESTS_SUCCESS
RESERVATION_REQUESTS_FAILURE
RESERVATION_REQUESTS_ERROR
IC_COMMIT_REQUESTS
COMMIT_REQUESTS_SUCCESS
COMMIT_REQUESTS_FAILURE
COMMIT_REQUESTS_ERROR
IC_DELETION_REQUESTS
DELETION_REQUESTS_SUCCESS
DELETION_REQUESTS_ERROR

Extension register
None

Associated logs
None

NUM_HalfCall_Success

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of successful half calls set up by the Policy Controller.

Associated registers
IC_RESERVATION_REQUESTS
RESERVATION_REQUESTS_SUCCESS
IC_COMMIT_REQUESTS
COMMIT_REQUESTS_SUCCESS
IC_DELETION_REQUESTS
DELETION_REQUESTS_SUCCESS

Extension register
None

Associated logs
None

SPC_TOPO_UPDATE

Description

Policy Controller Web Provision Topology Update
(SPC_TOPO_UPDATE)

OM group SPC_TOPO_UPDATE is a static OM. It provides registers for recording successful or failed topology update events in the Policy Controller Topology Manager.

The following table lists the key and info fields associated with OM group SPC_TOPO_UPDATE.

Key field	Info field
None	None

Related functional groups

Policy Controller static OM groups have only one tuple. The tuple key is 0 and the tuple name is empty.

Registers

The following table lists the registers associated with OM group SPC_TOPO_UPDATE and what they measure.

Registers for OM group SPC_TOPO_UPDATE

Register name	Measures
IC_PROV_TOPO_UPDATE	Total number of Topology Manager updates from Web provisioning
IC_PROV_TOPO_UPDATE_FAILURE	Total number of failed Topology Manager updates from Web provisoning
IC_OSS_TOPO_UPDATE	Total number of Topology Manager updates from OSS
IC_OSS_TOPO_UPDATE_FAILURE	Total number of failed Topology Manager updates from OSS

IC_PROV_TOPO_UPDATE

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the number of incoming topology updates from Web provisioning received by the Topology Provision Manager.

Associated registers

None

Extension register

None

Associated logs

SPCP603
TPM601

IC_PROV_TOPO_UPDATE_FAILURE

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the number of failed updates to the Topology Manager from Web provisioning.

Associated registers

None

Extension register

None

Associated logs

TPM601

IC_OSS_TOPO_UPDATE

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the number of incoming topology updates received by the Topology Provision Manager from OSS.

Associated registers

None

Extension register

None

Associated logs
SPCP603
TPM601

IC_OSS_TOPO_UPDATE_FAILURE

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the number of failed updates to the Topology Manager using OSS.

Associated registers
None

Extension register
None

Associated logs
TPM601

CAC_PER_NETWORK_ZONE

Description

Call Admission Control Request per Network Zone (CAC_PER_NETWORK_ZONE)

OM group CAC_PER_NETWORK_ZONE is the only per-instance OM. It records Call Admission Control requests sent by the Policy Controller to each Network zone.

The following table lists the key and info fields associated with OM group CAC_PER_NETWORK_ZONE.

Key field	Info field
None	None

Related functional groups

All registers in OM group CAC_PER_NETWORK_ZONE are associated with dynamically allocated tuple, based on assigned tuple key numbers and tuple key names.

Registers

The following table lists the registers associated with OM group CAC_PER_NETWORK_ZONE Sand what they measure.

Registers for OM group CAC_PER_NETWORK_ZONE

Register name	Measures
CAC_REQUEST	Total number of CAC requests that the Policy Controller sends to each network zone
CAC_SUCCESS	Total number of successful CAC requests that the Policy Controller sends to each network zone
CAC_FAILURE	Total number of failed CAC requests that the Policy Controller sends to each network zone

CAC_REQUEST

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the number of CAC requests that the Policy Controller sends to each Network zone.

Associated registers
None

Extension register
None

Associated logs
None

CAC_SUCCESS

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of successful CAC requests that the Policy Controller sends to each Network zone.

Associated registers
None

Extension register
None

Associated logs
None

CAC_FAILURE

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of failed CAC requests that the Policy Controller sends to each Network zone.

Associated registers
None

Extension register
None

Associated logs
SPCP 602

SPC_CMMT_REQ

Description

Policy Controller Commit Requests (SPC_COMMT_REQ)

OM group SPC_COMMT_REQ is a static OM. It records commit request events on the Policy Controller signaling interface.

The following table lists the key and info fields associated with OM group SPC_COMMT_REQ.

Key field	Info field
None	None

Related functional groups

Policy Controller static OM groups have only one tuple. The tuple key is 0 and the tuple name is empty.

Registers

The following table lists the registers associated with OM group SPC_COMMT_REQ and what they measure.

Registers for OM group SPC_COMMT_REQ

Register name	Measures
IC_COMMIT_REQUESTS	Total number of incoming commit requests received by the Policy Controller
COMMIT_REQUESTS_SUCCESS	Total number of commit success responses sent by the Policy Controller
COMMIT_REQUESTS_FAILURE	Total number of commit failure responses sent by the Policy Controller when a resource is unavailable
COMMIT_REQUESTS_ERROR	Total number of commit error responses sent by the Policy Controller because the request has an incorrect context

IC_COMMIT_REQUESTS

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the number of incoming commit requests received by the Policy Controller.

Associated registers

None

Extension register

None

Associated logs

None

COMMIT_REQUESTS_SUCCESS

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the total number of commit success responses sent by the Policy Controller.

Associated registers

None

Extension register

None

Associated logs

None

COMMIT_REQUESTS_FAILURE

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the total number of commit failure responses sent by the Policy Controller when a resource is unavailable.

Associated registers

None

Extension register

None

Associated logs

SPCP 602

COMMIT_REQUESTS_ERROR

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the total number of commit error responses sent by the Policy Controller because the request has an incorrect context.

Associated registers

None

Extension register

None

Associated logs

SPCP604

SPC_RSRV_REQ

Description

Policy Controller Reservation Requests (SPC_RSRV_REQ)

OM group SPC_RSRV_REQ is a static OM. It records reservation request events on the Policy Controller signaling interface.

The following table lists the key and info fields associated with OM group SPC_RSRV_REQ.

Key field	Info field
None	None

Related functional groups

Policy Controller static OM groups have only one tuple. The tuple key is 0 and the tuple name is empty.

Registers

The following table lists the registers associated with OM group SPC_RSRV_REQ and what they measure.

Registers for OM group SPC_RSRV_REQ

Register name	Measures
IC_RESERVATION_REQUESTS	Total number of incoming reservation requests received by the Policy Controller
RESERVATION_REQUESTS_SUCCESS	Total number of reservation success responses sent by the Policy Controller
RESERVATION_REQUESTS_FAILURE	Total number of reservation failure responses sent by the Policy Controller when a resource is unavailable
RESERVATION_REQUESTS_ERROR	Total number of reservation error responses sent by the Policy Controller because the request has an incorrect context

IC_RESERVATION_REQUESTS

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the number of incoming reservation requests received by the Policy Controller.

Associated registers

None

Extension register

None

Associated logs

None

RESERVATION_REQUESTS_SUCCESS

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the total number of reservation success responses sent by the Policy Controller.

Associated registers

None

Extension register

None

Associated logs

None

RESERVATION_REQUESTS_FAILURE

Register type

Peg type, double precision, up to 4,294,967,296 counts

Description

This register counts the total number of reservation failure responses sent by the Policy Controller when a resource is unavailable.

Associated registers

None

Extension register
None

Associated logs
SPCP 602

RESERVATION_REQUESTS_ERROR

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of reservation error responses sent by the Policy Controller because the request has an incorrect context.

Associated registers
None

Extension register
None

Associated logs
SPCP604

SPC_DEL_REQ

Description

Policy Controller Deletion Requests (SPC_DEL_REQ)

OM group SPC_DEL_REQ is a static OM. It records deletion request events on the Policy Controller signaling interface.

The following table lists the key and info fields associated with OM group SPC_DEL_REQ.

Key field	Info field
None	None

Related functional groups

Policy Controller static OM groups have only one tuple. The tuple key is 0 and the tuple name is empty.

Registers

The following table lists the registers associated with OM group SPC_DEL_REQ and what they measure.

Registers for OM group SPC_DEL_REQ

Register name	Measures
IC_DELETION_REQUESTS	Total number of incoming deletion requests received by the Policy Controller
DELETION_REQUESTS_SUCCESS	Total number of deletion success responses sent by the Policy Controller
DELETION_REQUESTS_ERROR	Total number of deletion error responses sent by the Policy Controller because the request has an incorrect context

IC_DELETION_REQUESTS

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the number of incoming deletion requests received by the Policy Controller.

Associated registers
None

Extension register
None

Associated logs
None

DELETION_REQUESTS_SUCCESS

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of deletion success responses sent by the Policy Controller.

Associated registers
None

Extension register
None

Associated logs
None

DELETION_REQUESTS_ERROR

Register type
Peg type, double precision, up to 4,294,967,296 counts

Description
This register counts the total number of deletion error responses sent by the Policy Controller because the request has an incorrect context.

Associated registers
None

Extension register
None

Associated logs
None