Nortel Multiservice Switch 7400

Operations: HDLC Transparent Data Service

NN10600-770

Document status: Standard
Document issue: 7.1S1
Document date: October 2005
Product release: PCR7.1 and up
Job function: Operations
Type: NTP
Language type: U.S. English

Contents

What’s new

There were no new features added to this document.

Attention: To ensure that you are using the most current version of an NTP, check the current NTP list in NN10600-000 *Nortel Multiservice Switch 7400/ 15000/20000 What’s New*.

HDLC Transparent Data Service configuration

Configure the HDLC Transparent Data Service to transmit data using any protocol (including proprietary protocols) that use HDLC at the link layer.

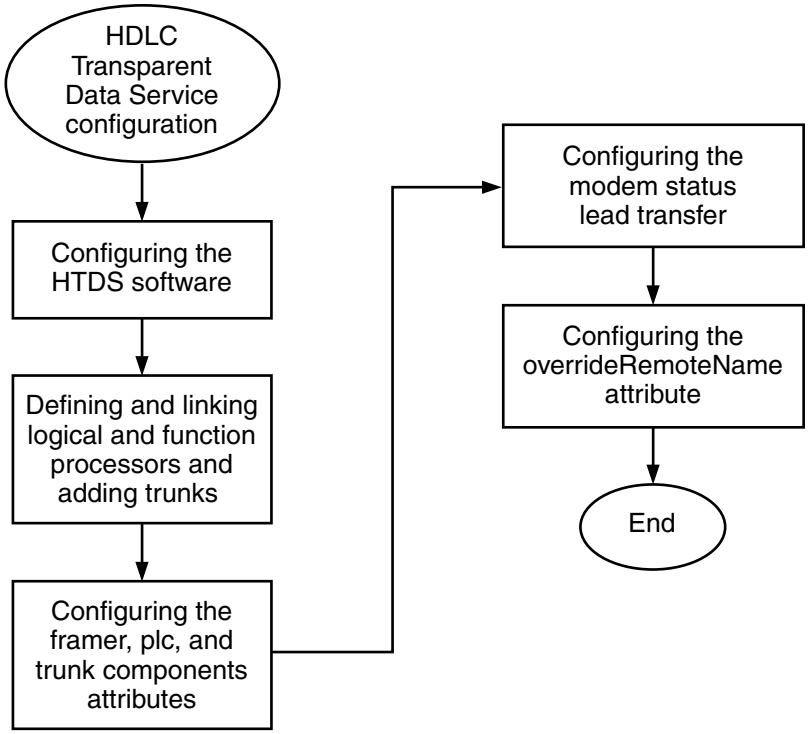
Prerequisites to HDLC Transparent Data Service configuration

- The version of your routing software contains the Path Oriented Routing System. The HDLC Transparent Data Service (HTDS) will not run without PORS. Refer to NN10600-435 *Nortel Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System* for information on how to provision PORS.
- All nodes in the network that are candidates for HTDS traffic is running the latest software and contains the *Trunk PathAdministrator* component to the view.

HDLC Transparent Data Service configuration procedures

This task flow shows you the sequence of procedures you perform to configure the HDLC Transparent Data Service. To link to any task or procedure, go to HDLC Transparent Data Service configuration procedure navigation (page 7).

HDLC Transparent Data Service configuration procedures



MSS 3428 006 AA

HDLC Transparent Data Service configuration procedure navigation

Configuring the HTDS software

Configure HTDS software on the two Nortel Multiservice Switch nodes that will run the HDLC Transparent Data Service.

Procedure steps

Step	Action
1	Define an instance of the *lpt* attribute. **`add software lpt/<name>`**
2	Add the HTDS application. **`set software lpt/<name> featureList <feature>`**
3	Repeat step 1 and step 2 to configure HTDS software on the remote end node.

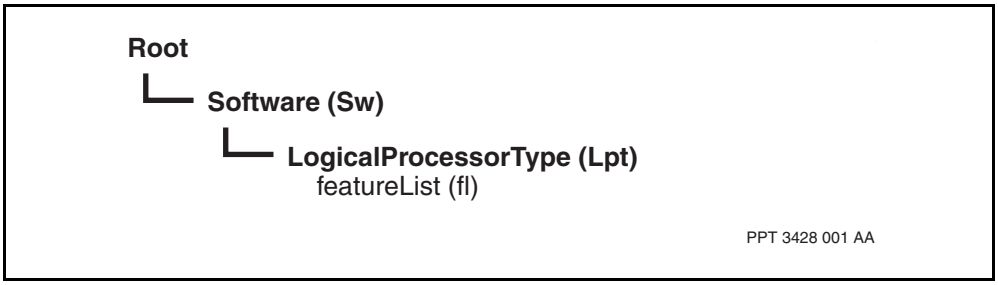
--End--

Variable definitions

Variable	Value
<feature>	is the value for the featureList attribute, either vtds or hdlcTransparent. The value vtds includes the HTDS, Voice Transport and Bit Transparent Data Service applications. The value hdlcTransparent provides only the HTDS application.
<name>	is the name for the lpt attribute, up to 25 ASCII characters. To help simplify provisioning, use a descriptive word, such as TDS.

Procedure job aid

HTDS software component hierarchy



Defining and linking logical processors and adding trunks

Define logical processors (LP) and link them to the configured HTDS software, then add trunks.

Procedure steps

Step	Action
1	Add logical processors. **`add lp/<lp_number>`**
2	Link the LPs to the configured HTDS software. **`set lp/<lp_number> lpt software lpt/<name>`**
3	Propagate the new settings throughout the module. **`activate prov`**
4	Define the HDLC Transparent Data Service instance. **`add htds/<htds_number>`**
5	Define the *interfaceName* attribute to link the *Framer* component to the hardware. **`set htds/<htds_number> framer interfaceName lp/ <lp_number> <port_type>/<port_number>`**
6	Define the *Plc* component's *remoteName* attribute. **`set htds/<htds_number> plc remoteName <remote_name>`**
7	Add a *Trunk* component instance, if necessary. **`add trunk/<trunk_number>`**
8	Add the *PathAdmin* component. **`add trunk/<trunk_number> PathAdmin`**
9	Repeat step 1 through step 8 to define and link logical processors and to add trunks to the other end of the connection.

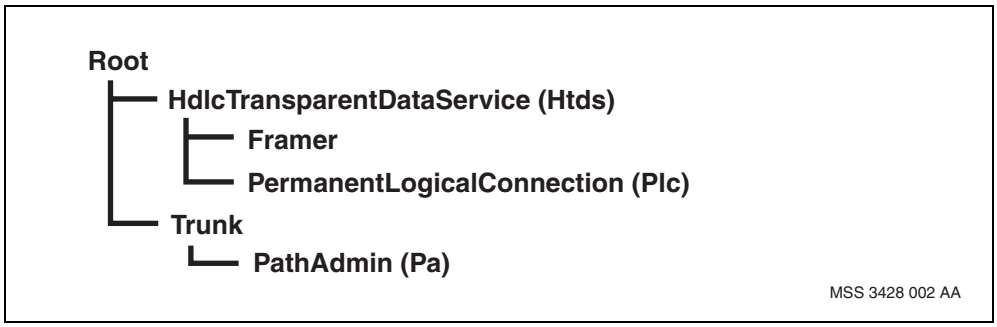
--End--

Variable definitions

Variable	Value
<htds_number>	is the instance number of the Htds component. You can assign any value in the range as long as it does not exist anywhere else on the node.
<lp_number>	is the logical processor value that you assigned when you provisioned the card.
<name>	is the name you assigned to the HTDS software (for example, TDS).
<port_number>	is the port number that you assigned when you provisioned the port.
<port_type>	is the type of port, either X21 (for a V.11 port) or V35 (for a V.35 port).
<remote_name>	is the address of the PLC remote end-point (the name of the other end of the connection.) End points are identified using a node name and a service name. For a connection to be established, remote names between end points must be consistent. The value for the *remoteName* attribute must match the node name exactly or a connection will not be made.
<trunk_number>	is the instance number of the Trunk component. See NN10600-420 *Nortel Multiservice Switch 7400/15000/20000 Operations: Trunking* for more information about adding a Nortel Multiservice Switch trunk.

Procedure job aid

HTDS component hierarchy



Configuring the attributes of the Framer, Plc, and Trunk components

Specify some of the parameters that PORS uses in creating the connection by configuring the attributes of the Framer, Plc, and Trunk components.

In some cases, changes to the attributes under the *Plc* component require accompanying changes to the attributes under the *Trunk* component in order to work.

Prerequisites

- Before you use these attributes, be sure that you understand how they will affect your network. For explanations of what these attributes do, see NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference*.

Procedure steps

Step	Action
1	Define values for the attributes that you want to change for the *Framer* component. You can change some or all of the attributes. **`set htds/<htds_number> framer <framer_attribute> <framer_attribute_value>`**
2	Define values for the attributes that you want to change from the default values to some other value for the *Plc* component. **`set htds/<htds_number> plc <plc_attribute> <plc_attribute_value>`**
3	Define values for the *Trunk* component attributes for which you want to specify values. **`set trunk/<trunk_number> pathAdmin <trunk_attribute> <trunk_attribute_value>`**
4	Repeat step 1 through step 3 to configure the attributes of the Framer, Plc, and Trunk components at the other end of the connection.

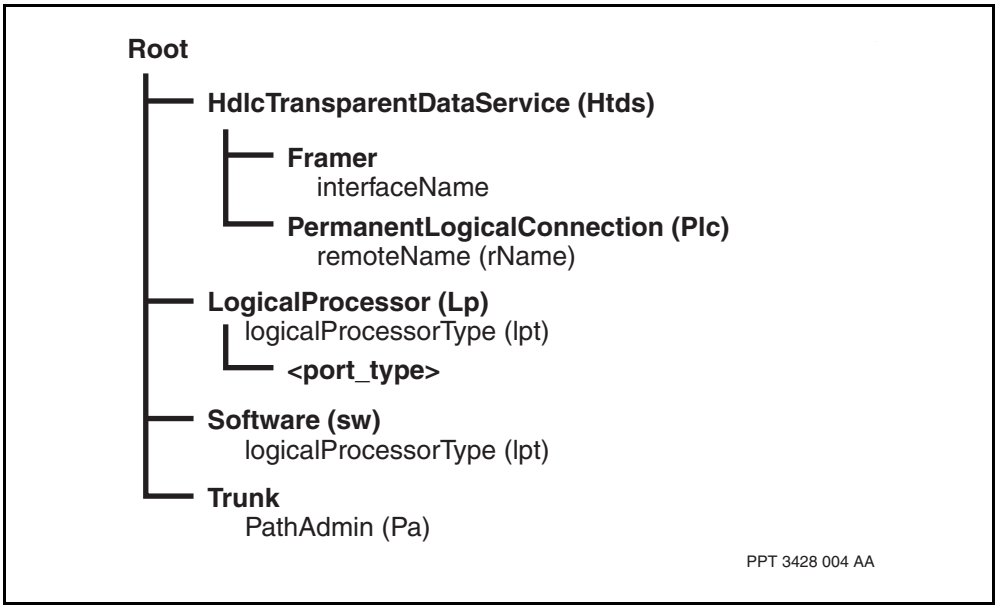
--End--

Variable definitions

Variable	Value
<framer_attribute>	is a provisionable attributes of the *Framer* component.
<framer_attribute_value>	*is the value of the Framer component's provisionable attribute.*
<htds_number>	is the instance number of the Htds component.
<plc_attribute>	is a provisionable attributes of the *Plc* component.
<plc_attribute_value>	is a value of the *Plc* component's provisionable attribute. Set the values for the *requiredTxBandwidth* and *requiredRxBandwidth* attributes to reflect the amount of data that will be sent into the network.
<trunk_attribute>	is a provisionable attributes of the *Trunk* component.
<trunk_attribute_value>	is the value of the Trunk component's provisionable attribute.
<trunk_number>	is the instance number of the trunk.

Procedure job aid

Framer, Plc, and Trunk component hierarchy



Configuring modem status lead transfer

Configure modem status lead transfer to send status lead changes transparently through the network.

Procedure steps

Step	Action
1	Configure a port to ignore its logical hardware attribute *readyLineState*. **set lp/<lp_number> <port_type>/<port_number> readyLineState !**
2	Configure a port to ignore its logical hardware attribute *dataTransferLineState*. **set lp/<lp_number> <port_type>/<port_number> dataTransferLineState !**
3	Set one end of the subnet to DCE. **set lp/<lp_number> <port_type>/<port_number> linkmode dce**
4	Set the other end of the subnet to DTE. **set lp/<lp_number> <port_type>/<port_number> linkmode dte**
5	Ensure that the time delay between the detection of a modem status lead change and the actual sending of the status lead change is set for the port. **set lp/<lp_number> <port_type>/<port_number> lineStatusTimeOut 500**
6	Set the *lineSignalTransport* attribute to on to ensure that modem status lead changes are sent transparently through the network so that changes occurring at one end of the network are reflected at the other end. **set htds/<htds_number> framer lineSignalTransport on**

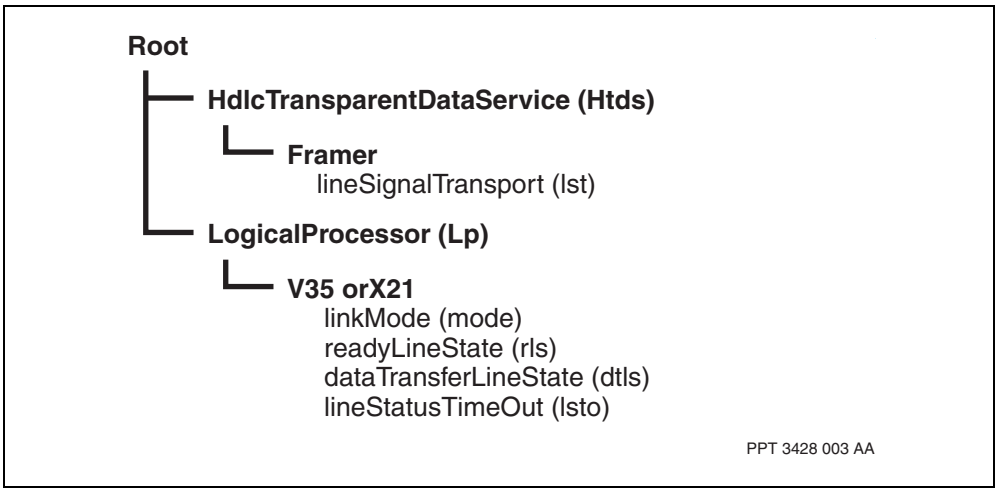
--End--

Variable definitions

Variable	Value
<htds_number>	is the instance number of the *Htds* component.
<lp_number>	is the logical processor value that you assigned when you provisioned the card.
<port_number>	is the port number that you assigned when you provisioned the port.
<port_type>	is the port type, either X21 (for a V.11 port) or V35 (for a V.35 port).

Procedure job aid

Modem status lead transfer component hierarchy



Configuring the overrideRemoteName attribute

Set the *overrideRemoteName* attribute to override the setting for the *remoteName* attribute.

Procedure steps

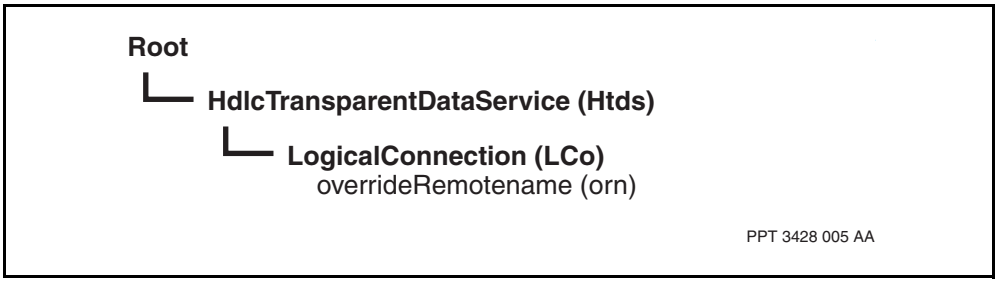
Step	Action
1	Set the *overrideRemoteName* attribute. **`set htds/<htds_number> lc overrideRemoteName <override_name>`**
	--End--

Variable definitions

Variable	Value
<htds_number>	is the instance number of the Htds component.
<override_name>	is an override name for the address of the PLC remote end-point.

Procedure job aid

overrideRemoteName attribute component hierarchy



Troubleshooting

Use this chapter to learn about what steps you can take to solve problems that may occur after you have installed HTDS. Also, learn about how to use provisioning checklists to help ensure a successful HTDS setup.

Navigation

Provisioning checklists

To avoid having problems establishing a connection, please use the following checklists:

Installation

Make sure you consider the following:

- Do you have the correct software level loaded on every Nortel Multiservice Switch 7400 node that could be a path candidate?
- Have you installed PORS and HTDS according to the instructions in NN10600-270 *Nortel Multiservice Switch 7400/15000/20000 Software Installation*?
- Have you added a *Trunk PathAdmin* component to every Multiservice Switch 7400 node that could be a path candidate?

Provisioning with default values

Make sure you consider done the following:

- Have you linked the *Framer* component to the hardware? Did you use the correct logical processor value and port number? Is the syntax correct?

- Did you use the *remoteName* attribute to identify the other end of the connection? Did you use the exact node name and correct syntax?
- Have you displayed your provisioning and checked it for errors?
- Have you used the check, save, activate, and confirm, commands?
- Have you provisioned both ends of the connection?

Provisioning optional features

Make sure you consider the following:

- Refer to the checklist in the section entitled Provisioning with default values (page 16). All of the entries in that section also apply to this section.
- Are you sure that the bandwidth is available? (Remember that you are probably sharing the total bandwidth with connectionless routing.)
- Are the provisioned attributes under the *Trunk Unack Framer* component identical at both ends of the connection? If they are not, you will not get a connection.
- Attributes in the same row are interdependent. Provision them accordingly.
- Several of the optional attributes have provisioning dependencies, see the table Optional Attributes (page 17).

Optional Attributes

PLC attributes	Trunk attributes	HDLC Framer attributes
		dataInversion If set to on, 0 is changed to 1 and 1 to 0 in data bits sent to the interface. Incoming data bits will be inverted before being processed.
		lineSignalTransport Allows modem status leads to be transferred transparently through the network on either V.11 or V.35 function processors.
		frameCrcType Allows you to specify the type of cyclic redundancy check (CRC) used or to specify that no CRC should be used.
(1 of 4)		

Optional Attributes (continued)

PLC attributes	Trunk attributes	HDLC Framer attributes
		flagsBetweenFrames This attribute defines the number of flags that are inserted between frames sent to the link interface.
		nonOctetData If set to no, the incoming data must be octet-aligned. If set to yes, non-octet data can be received, but CRC checking will not be operational.
	framingType Allows highest priority bit cells to use the interrupt queue if it is set to interrupt. This tool must be set to the same value on both ends of the connection.	
	maxLc Defines the maximum number of logical channels that can exist on a trunk simultaneously. A path will not be set up on a trunk if this number is exceeded.	
	reservedBwOut Allows you to reserve bandwidth for permanent logical connections. This bandwidth must be available if you are to get a connection.	
setupPriority Defines the ability of a new path to bump an existing path.		
holdingPriority Defines the ability of an existing path to maintain a path and not be bumped by a new path.		
requiredTxBandwidth		
requiredRxBandwidth		
(2 of 4)		

Optional Attributes (continued)

PLC attributes	Trunk attributes	HDLC Framer attributes
requiredTrafficType The value specified must be a member of those defined by the supported trunk attribute.	*supportedTrafficTypes* Must include the value defined in the PLC attribute if a trunk is to become part of the path.	
permittedTrunkTypes The value specified must include a specific trunk type if a trunk of that type is to be used in the path.	*trunkType* Defines the type of trunk.	
requiredSecurity All trunks in the path will have at least the security level selected.	*trunkSecurity* Defines the security level of a particular trunk.	
requiredCustomerParameter All trunks in the path will have at least the value selected.	*customerParameter* Defines a value for a particular trunk.	
pathAttributeToMinimize Choose cost or delay but not both.		
maximumAcceptableCost The sum of the corresponding trunk attribute values will not exceed this figure. The sum taken over all trunks in selected route.	*trunkCost* Allows you to define a value for a trunk.	
maximumAcceptableDelay The sum of the trunk delay values will not exceed this figure.		
emissionPriority Defines which cells or frames are sent first. This has important implications for network engineering.		
(3 of 4)		

Optional Attributes (continued)

PLC attributes	Trunk attributes	HDLC Framer attributes
discardPriority Defines which cells or frames are discarded first if congestion occurs. This has important implications for network engineering.		
optimization Determines if the connection should attempt to follow through with the optimization process when requested by PORS connection control.		
bumpPreference This attribute controls when bumping occurs in the route selection process. By default, bumping occurs only when necessary. In other words, a connection will bump only if there is no other way for the connection to be established. With this attribute, a connection can be set to bump in order to get its best route.		
(4 of 4)		

HTDS-related alarms

The alarms related to HTDS are as follows:

- 7018 0001 to 7018 0004 Path Administrator-related alarms
- 7018 1001 and 7018 1002 LCo-related alarms
- 7019 0001 HTDS-related alarm

For information on these alarms, refer to NN10600-500 *Nortel Multiservice Switch 6400/7400/15000/20000 Alarms Reference*.

Problem solving

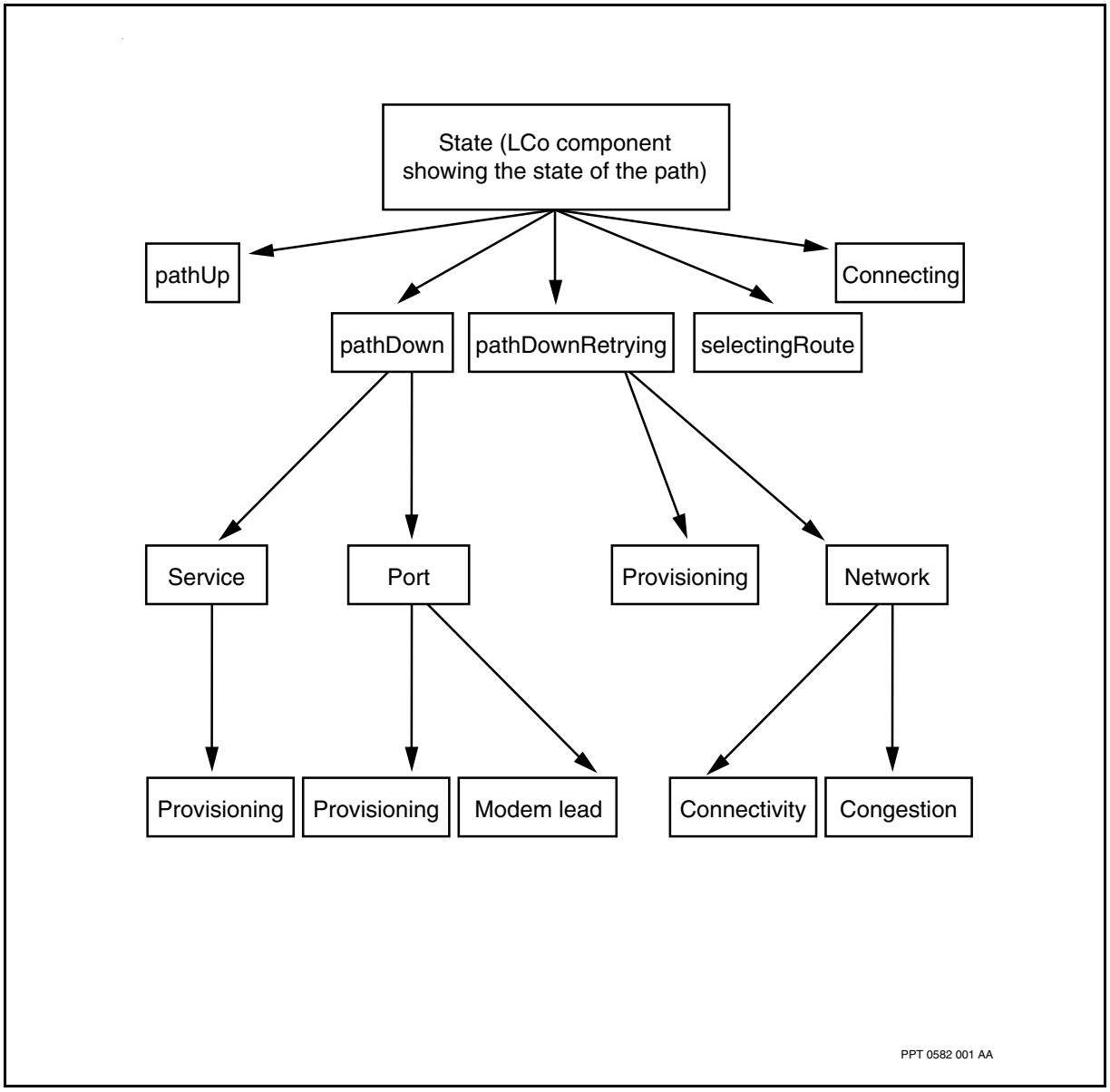
Problems with setting up connections may be due to errors or mismatches in setting up the system or provisioning. Use the information in Provisioning checklists (page 16) to check if there are any steps that you may have forgotten in the process. Remember even simple spelling errors can cause provisioning mismatches.

In general, you should look for the following:

- is the path up or down?
- where does it go down?
- why did it go down?

To help determine the answer to these questions, see the figure Flowchart: an example of troubleshooting using the LCo component (page 21). See table Handling problems (page 22) to help you determine how to respond to an occurring problem.

Flowchart: an example of troubleshooting using the LCo component



Handling problems

Problems that may occur	Probable causes	Corrective measures
Unable to provision HTDS.	Error in card provisioning.	Check the card. If an orange light is glowing, check the values for the *logicalProcessor* and *logicalProcessorType* attributes. Reprovision using correct values for the *lp* and *lpt* attributes. For provisioning information see NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference* and NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures*.
The connection doesn't come up.	Physical connections are not secure.	Check the cable, connections, and pins. See NN10600-175 *Nortel Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*. For a description of an FP see NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description*. To install an FP see NN10600-175 *Nortel Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*. To provision an FP see NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures* and NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference*.
	Error in provisioning the service.	Check the service provisioning for both ends of the connection. See Provisioning checklists (page 16). Trunk provisioning information must be identical for both ends of the connection. Reprovision with correct information.
The connection doesn't come up (continued).	Specified bandwidth is not available.	Check the available bandwidth of trunks in the path. Reprovision using less bandwidth if it is not needed or re-engineer the network to make bandwidth available.
	If the *manualPath* attribute has been used, one of the nodes or trunks used may not be operational.	Check nodes/trunks for failure. Reprovision using a path that does not include failed nodes or trunks.
	Error in port provisioning.	Check port provisioning. See NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference* and NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for provisioning information.
Attention: If a problem has occurred before with this connection, check the *lastTeardownReason* attribute under the *Plc* component.		

(1 of 2)

Handling problems (continued)

Problems that may occur	Probable causes	Corrective measures
The PLC is up and running, but no data is being sent.	DCE–DTE is not provisioned properly on the subscriber's end.	Check the subscriber's end-DCE–DTE provisioning.
	User's-end terminal may be experiencing problems.	Check the terminal. Take appropriate action to rectify the problem.
	Access line to Multiservice Switch 7400 may not be transmitting data.	Check the access line. Rectify any problems encountered.
Connection goes down and doesn't reset.	Under extreme circumstances (for example no suitable trunk is available) a path may take 1–2 minutes to reroute.	Wait 1–2 minutes and check to see if rerouting has occurred.
	If the security option is being used, no sufficiently secure trunk may be available.	Check trunk provisioning. Take appropriate action to see that a secure trunk is made available.
Unexpected data loss.	Congestion.	Check bandwidth utilization. Take steps to reduce congestion if the problem continues.
	Poor trunk error performance.	See NN10600-420 *Nortel Multiservice Switch 7400/ 15000/20000 Operations: Trunking*.
Trunk does not achieve locked state.	Operator did not allow for expected delay.	The trunk may remain in the shutting down state for up to 30 seconds before achieving the locked state.
Modem status lead changes not recognized by remote end.	Time delay for debounce set to a high value.	Check the provisioned value for the *lineStatusTimeOut* attribute. Reprovision attribute to a lower value.
	Improper setting of modem status lead attributes.	Check provisioning attributes in Configuring modem status lead transfer (page 13).

(2 of 2)

Troubleshooting examples

In the examples given here, the operator has provisioned a route that has failed to come up. The operator looks at the Routing RouteSelector component's *reasonForNoRoute* attribute to determine the reason.

Example 1

```
        d rtg rs

Rtg Rs
selectedRouteDescription =
routeCostMetric          = 0
routeDelayMetric         = 0 ms
reasonForNoRoute         = unknownRemoteNodeName
routeSelectionAttributes = fromOperator
sourceId                 = 1157
remoteName               = /NOdeR2b
setupPriority            = 4
requiredTxBandwidth      = 16000 bit/s
requiredRxBandwidth      = 16000 bit/s
maximumTransmissionUnit = 0
security                 = 4
trafficType              = data
permittedTrunkTypes      = terrestrial satellite tt1 tt2 tt3
~tt4 ~tt5 ~tt6
customerParameter        = 4
minimizationCriterion    = cost
maximumAcceptableCost    = 1280
maximumAcceptableDelay   = 100000 m
statistics               =                  statisticsTable

                 setupPriority|    0    1    2   3    4
                --------------+ ---- ---- --- ---- ----
                routesRequested| 5788 5785 419 5786 6106
                 routesSelected| 3275 3273 166 3275 3309

ok
```

The *reasonForNoRoute* attribute indicates that the value specified for the *remoteName* attribute is unknown. In this case the value for the *remoteName* attribute used is also displayed; for other types of problems, the operator may need to display another attribute to show the required information.

Example 2

```
        d rtg rs

Rtg Rs
selectedRouteDescription =
routeCostMetric          = 0
routeDelayMetric         = 0 ms
reasonForNoRoute         = unknownRemoteNodeName
```

```
routeSelectionAttributes = fromLastRouteRequest
sourceId                 = 1157
remoteName               = /BaseOne
setupPriority            = 4
requiredTxBandwidth      = 16000 bit/s
requiredRxBandwidth      = 16000 bit/s
maximumTransmissionUnit = 0
security                 = 4
trafficType              = data
permittedTrunkTypes      = terrestrial satellite tt1 tt2 tt3
~tt4 ~tt5 ~tt6
customerParameter        = 4
minimizationCriterion    = cost
maximumAcceptableCost    = 1280
maximumAcceptableDelay   = 100000 ms
statistics               =                  statisticsTable

                  setupPriority|   0    1   2    3    4
                ---------------+ ---- ---- --- ---- ----
                routesRequested| 5785 5782 419 5783 6103
                routesSelected|  3275 3273 166 3275 3309
```

In this case, the problem can be identified as a typographic error in the provisioning for the *remoteName* attribute. The / mark at the beginning of the node name is incorrect. To correct the problem, the operator would reprovision the *remoteName* attribute with the correct name of the other end of the connection.

HDLC Transparent Data Service fundamentals

Use this chapter to learn about the basic functionality of the Nortel Multiservice Switch HDLC Transparent Data Service.

Navigation

What is the HDLC Transparent Data Service?

The HDLC Transparent Data Service (HTDS) provides a means of transmitting data using any protocol (including proprietary protocols) that use HDLC at the link layer. Proprietary protocols do not present problems since the service does not process the data. The frames are routed unchanged (transparently) by means of a end-to-end connection.

HTDS on a Nortel Multiservice Switch 7440 network can transmit X.25 protocols, for example. HTDS provides efficient bandwidth usage. The service only occupies network resources when there is data being transmitted. Idle patterns (such as flags) are suppressed at the source and regenerated at the destination.

This transmission scheme offers good transfer speeds, low bandwidth requirements, and low costs. The service is provided through the use of common Multiservice Switch trunks, and usually without the need for additional customer equipment.

The connections are provided by the Path Oriented Routing System (PORS) in the form of dynamically determined fixed paths that allow minimal delay variations within a connection. PORS maintains ordering but does not guarantee delivery. Frame recovery is left to higher layer protocols.

Service characteristics

The HDLC Transparent Data Service's key characteristics are

- bidirectional network-wide transfer of octet data and non-octet data
- support of Non-return to Zero (NRZ) encoding (Non-return to Zero Inverted (NRZI) is not supported)
- suppression of idle flags; saving network bandwidth
- provisionable attributes, including
  - setup, holding, emission, and discard priorities
  - transmit and receive bandwidth
  - cost or delay minimization
  - action on path failure
  - cyclic redundancy check (CRC), flags between frames, and dataInversion
  - support of octet or non-octet aligned data
- service interconnections
  - V.11 to V.11 or V.35 to V.35
  - DS1 to DS1
  - E1 to E1
  - V.11 or V.35 to DS1 or E1
  - V.11 to V.35
  - DS3 to DS3
  - DS3 to DS1
  - HSSI to HSSI
  - HSSI to DS1
  - HSSI to DS3

System requirements of the HDLC Transparent Data Service include

- Path Oriented Routing System
- function processor (FP) cards. For information on which FPs support HTDS, refer to NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference*.

For more related information refer to the following documents:

- NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description*
- NN10600-175 *Nortel Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*
- NN10600-551 *Nortel Multiservice Switch 7400/15000/20000 FP Configuration Reference*
- NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures*
- NN10600-500 *Nortel Multiservice Switch 6400/7400/15000/20000 Alarms Reference*

Other characteristics of the HDLC Transparent Data Service

This section discusses the following topics:

- transport of HDLC frames transparently
- resource management
- transport of modem status leads

Transport of HDLC frames transparently

HDLC frames are formed into packets at the source node. Each packet contains the original HDLC frame and all path information required to reach the destination. The HDLC data portion of each packet is transferred across the network transparently (that is, without being interpreted). Only idle flags are removed for greater efficiency.

Resource management

To provide HDLC service, each end-point HTDS application provides the following functions:

- handling of service-provisioning data and commands
- creation and management of service resources, such as components and processes to:
  - establish the end-to-end path during the service session
  - handle operator requests to monitor the service (for example, trace the path or collect statistics on performance characteristics)
  - aid in managing congestion
  - aid with alarm and log duties

Transport of modem status leads

The modem status leads capability allows the status lead change at one end of the HTDS connection to be sent transparently through the network. This ensures the status lead change is reflected at the other end of the HTDS connection. Figure Modem status lead transfer in a Multiservice Switch network (page 30) depicts a Multiservice Switch 7440 network. When a modem status change occurs on User equipment A, this change is sent to node A propagated through the subnet, through node B then to User equipment B.

The modem status lead feature is used when the following conditions are met:

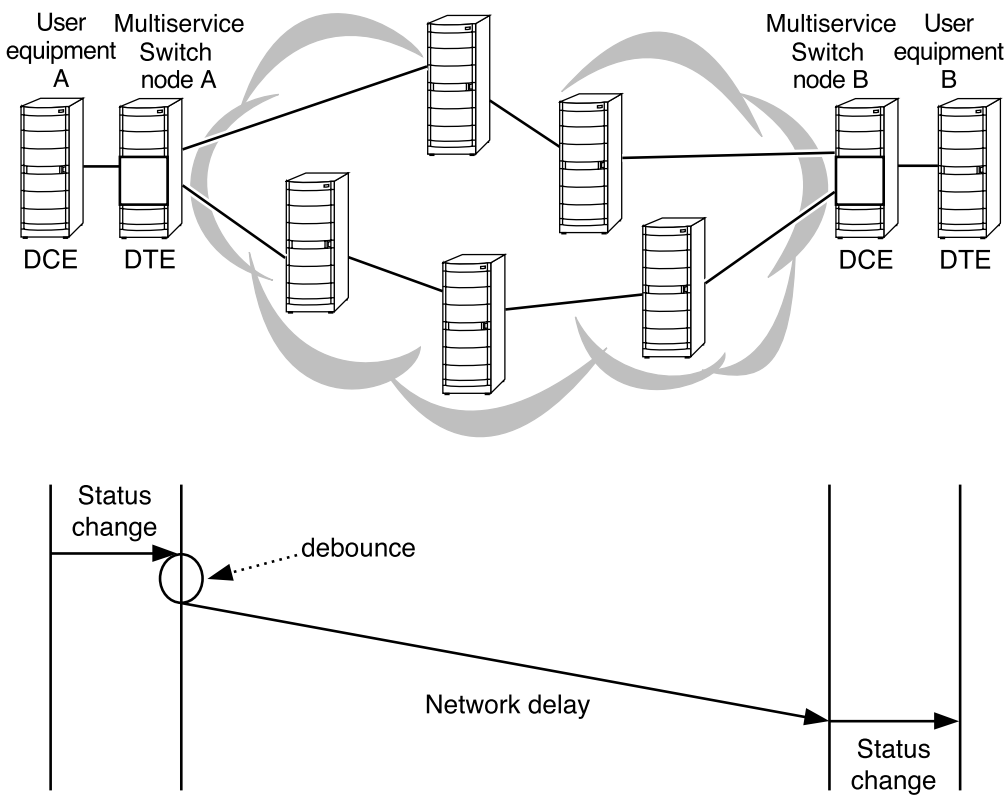
- the access points must be either all V.35 function processors or all V.11 function processors
- one end of the subnet is set to DTE and the other end is set to DCE
- the *lineSignalTransport* attribute under the *htds Framer* component is set to on

If the access points are not the same (for example node A has V.35 interfaces only and node B has V.11 interfaces only) or the *lineSignalTransport* attribute is set to off, then the leads are terminated at the local functional processor (in our case, in node A FP). Details on provisioning the modem status lead feature is contained in Configuring modem status lead transfer (page 13).

Modem status lead changes are sent asynchronously with respect to data. Although the order of status lead changes are preserved at the interface, the actual time periods between modem leads are not the same. The bottom portion of the figure Modem status lead transfer in a Multiservice Switch network (page 30), depicts the timing delays associated with the sample network. Timing delay can occur at two separate points, at the network interface, and through the network.

Timing delay can occur at two separate points, at the network interface, and through the network. Timing delay at the network interface is a function of the debounce setting. The setting for the debounce is dependent on the provisionable *lineStatusTimeOut* attribute for the V.11 and V.35 function processors. Setting of this attribute to a high value will result in timing delays between the detection of a modem status lead change and the actual sending of the status lead change. Timing delay through the network is dependent on the specific network traffic.

Modem status lead transfer in a Multiservice Switch network



Note: This feature is only supported when the access points (Multiservice Switch node A and B) each contain a V.35 function processor or each contain a V.11 function processor and one end of the Multiservice Switch subnet is set to DTE (Multiservice Switch node A) and the other end is set to DCE (Multiservice Switch node B).

Establishing connections

The figure Path and path description (page 32) depicts a path across a six-node network and illustrates some of the terms used in this section.

The Path Oriented Routing System sets up a Permanent Logical Connection (PLC) between two ends of a network path. The connection is called permanent since it is established through provisioning and, barring network difficulties, remains in place until removed. The fixed path is determined by the resources available when the path was established. Once the path has been provisioned, the HTDS user may regard it as an end-point to end-point wire.

The following sections which follow contain information about establishing connections across a network:

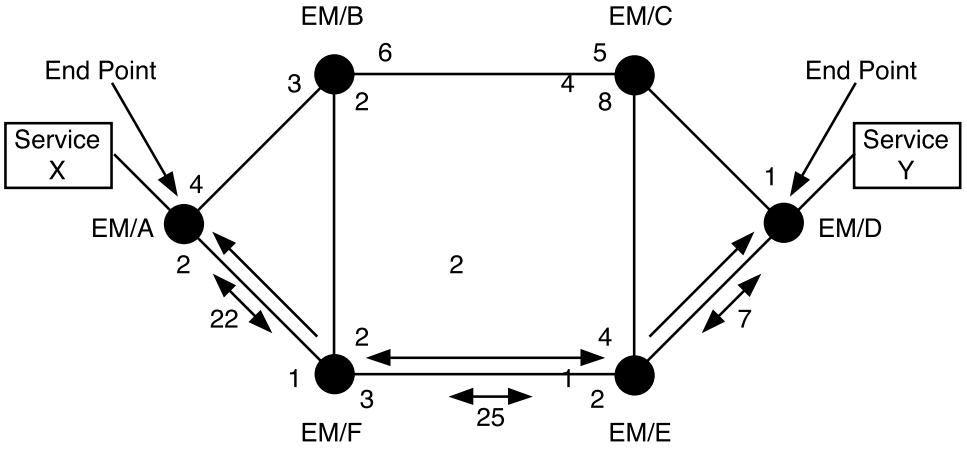
Establishing a path

To establish a path across the network, the end points must be uniquely identified. This identifier is established when you provision the service and is used to route packets to their destination. PORS and HTDS are provisioned together in one session.

To connect (make a path between) the two end points, you provision the name of the far-end HTDS component into the near-end HTDS component. Details are described in HDLC Transparent Data Service configuration (page 6). Path establishment is automatic when both ends of the of the connection have been successfully provisioned with the HTDS service. For more information about components, see NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference*.

Some provisioning data should either be identical at both ends of the connection (for example some of the attribute values under the *Framer* component) or must point to the exact identifier of the other end of the connection (the *remoteName* attribute under the *Plc* component). If the other end name is not correct, no path will be established.

Path and path description



Legend:

2 = Trunk instance

25 = LCh - Logical Channel Number

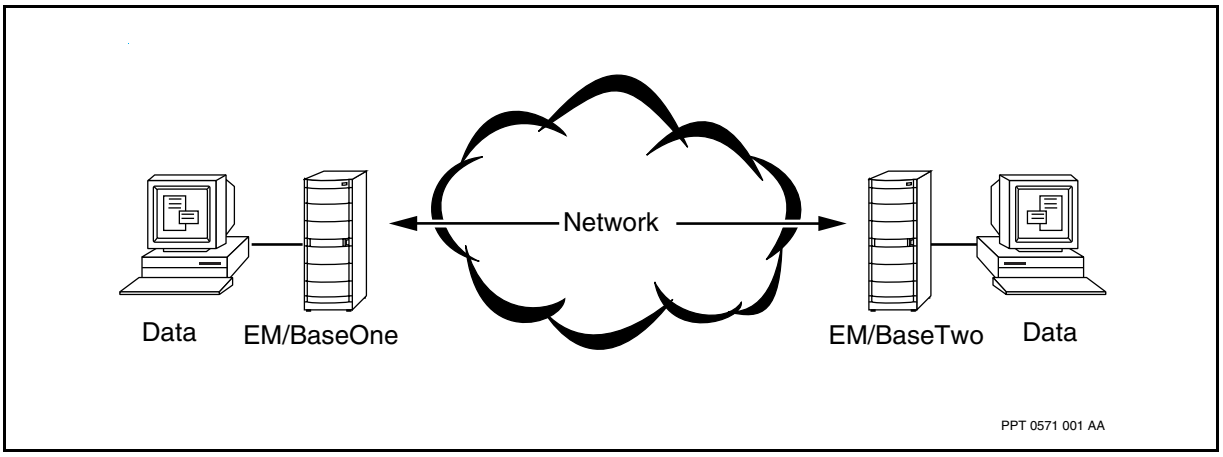
= Node

Path Description = EM/A Service/X, EM/A Trunk/2 LCh/22. EM/F Trunk/3 LCh/25, EM/E Trunk/1 LCh/7, EM/D Service/Y

Note: By convention, outbound trunk names are used to describe paths and routes. EM refers to Multiservice Switch-node enterprise module.

MSS 0570 001 AA

Multiservice Switch 7440 path across the network



Using default values

You do not necessarily need to provision most attributes associated with the service. HTDS comes with a set of default values for most of the attributes. The default values are designed to set up a Permanent Logical Connection using the optimal route across the network. The majority of networking applications should find the default values sufficient. It is a good idea to use the default values for the initial setup. Add options only as needed.

In many networks the default values are sufficient; however, you can choose how PORS selects a route. Attributes that allow selection are explained in the section entitled Route selection (page 39).

Creating the path

The path is established on a hop-by-hop basis. A set-up packet is sent down the route chosen by the Route Selector (RS). As the packet follows the route it uses the Nortel Multiservice Switch trunks that will be necessary to complete the path. At each point along the route the following actions are triggered:

- creation of the *Logical Channel* (*Lch*) components on all the trunks
- allocation of the Logical Channel Numbers (LCNs) to be used on each trunk
- verification of bandwidth availability
- reservation of bandwidth

When the path-setup packet reaches the destination end point, a path-setup confirmation packet is returned to the source end point in the reverse direction. This enables the path for data transfer.

Path bumping

Path bumping is the forced rerouting of an existing path by a new higher priority path of another logical connection. Bumping happens when there is not enough bandwidth in the network to establish a new path. The rerouting can in turn cause bumping of other paths. It may happen that a bumped path cannot be re-instantiated if the network is heavily loaded.

Optimizing paths

Over time, a PORS connection may end up on a less than optimal path due to link failures, node software upgrades, Nortel Multiservice Switch trunks being locked, or other possible scenarios. Path optimization periodically attempts to move the PORS connection back to a more optimal path. The first step of the optimization process begins when the routing system determines the best available path and compares it with the path currently used by the connection. If this new path provides better metrics, the connection is moved to the new path and the original path is released.

If the new path does not provide better metrics, path optimization will then attempt to balance the PORS load on the link groups used to carry the path. This process involves moving the connection (which is being optimized) to a new path established on a different link in the link group. This will only occur if it contributes to re-balancing the load on the link groups.

The optimization process is administered by PORS Connection Control which resides on each Multiservice Switch node in the network. For more information on path optimization, refer to NN10600-435 *Nortel Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*.

Path optimization is an optional feature that must be provisioned on a node. Enter the provisioning mode and type the add rtg pors command to activate this feature.

Recovering from path establishment failure

The selected path can fail to establish under the following conditions:

- there is not enough bandwidth available
- there is a failure (node, function processor, or Nortel Multiservice Switch trunk) along the chosen route
- the trunk has reached the maximum number of paths that it can support (*maxLc* attribute)

In the case of a failure, a path-setup failure packet is returned from the point of failure back to the source end point. The end point reports the failure reason to the RS and requests a new route. If another route is not available even with bumping, RS informs the end point that the path cannot be set up. If another route is selected by RS, the end point starts the path-setup procedure again.

Multiservice Switch trunk bandwidth allocation

This section lists provisionable parameters, provided by a Nortel Multiservice Switch trunk, that allow different policies of bandwidth allocation to be enforced. It is up to the network engineers to decide what constitutes an efficient sharing of resources.

Reserving bandwidth

Bandwidth on a Nortel Multiservice Switch trunk is shared between connectionless and connection-oriented traffic. Bandwidth that is unused by one traffic type can be used by the other. PORS reserves bandwidth in both directions on each trunk in the path. This reservation is not enforced by PORS but is used to determine the number and size of the paths that can be set up on a given trunk. Bandwidth is expressed in bit/s in each direction. Path instantiation on a trunk is delimited by Trunk Path Administrator-provisionable attributes *maxLc* and *MaxReservedBwOut*. Use the *requiredRxBandwidth* and *requiredTxBandwidth* attributes under the *Plc* component to reserve bandwidth for a path.

maxLc attribute

This is the limit on the number of individual Logical Channels (or paths) that traverse this Nortel Multiservice Switch trunk. When this number is reached, no new paths can be established over this trunk until some existing paths clear.

maxReservedBwOut attribute

This is the percentage of total Nortel Multiservice Switch trunk bandwidth which PORS can allocate among individual Logical Channels. Once this percentage is reached, the trunk has no more reservable bandwidth. No paths can establish over this trunk until some existing paths clear.

For example, on a DS-1 trunk using all timeslots at 1.536 Mbit/s, a value of 65% for this trunk attribute makes this trunk capacity appear to be 0.9984 Mbit/s for path-oriented routing. Connectionless traffic can use the remaining 0.5376 Mbit/s. Hence, PORS never reserves more than 0.9984 Mbit/s of this trunk.

Specifying setup and holding priorities (path bumping)

All PLCs in PORS have setup and holding priorities assigned to them. If a route with sufficient unreserved bandwidth cannot be found for a PLC, existing paths may be moved elsewhere to free up bandwidth. This process is called path bumping. Existing path-holding priorities and new path-setup priorities are compared to determine when a new path may bump an existing path. An attempt is made to reroute a path which has been bumped.

Setup and holding priorities accommodate scenarios where customers would like to determine which paths are allocated bandwidth at setup time (setup), but once set up, the paths have to remain (holding) to minimize disruption. For

example, if a network is carrying video through BTDS, voice through the voice service, and data through HTDS, and the user considers video to be the highest priority, data to be next, and voice to be the lowest, one way of accommodating such a requirement is as follows:

The table Example setup and holding priorities (page 36), lists some sample values that may be used.

Example setup and holding priorities

Traffic type	Setup priority	Holding priority
Data	Medium	Medium
Voice	Low	High
Video	High	High

The *setupPriority* and the *holdingPriority* attributes under the *Plc* component specify these priorities. A high holding-priority path will not be moved by a lower setup-priority path. Conversely, a high setup-priority path may bump lower holding-priority paths.

Each priority may have one of five values, ranging from zero (0) to four (4), where 0 is the most important path and has the highest priority, and 4 is the least important and has the lowest priority. A new path can bump an existing path only if the value of the *setupPriority* attribute for the new path is greater (that is, higher priority) than the value of the *holdingPriority* attribute for the existing path.

Setup and holding priority have a default setting of medium (2). PLCs of more, or of less, importance than the default can be reassigned other values.

For more details on path bumping, see NN10600-435 *Nortel Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*.

Specifying emission and discard priorities

Emission priority is a measure of how urgently a packet will be emitted to the Nortel Multiservice Switch trunk. The more urgent the emission priority, the faster the packet is sent to the trunk. Emission and discard priorities are set independently. HTDS frames generally do not tend to have a high emission priority and may or may not have a high discard priority according to the type of frames being sent.

Emission and discard priorities can have far-reaching implications for congestion management in your network. Do not adjust these values until you have considered all of the implications for network traffic.

The *emissionPriority* and *discardPriority* attributes under the *Plc* component affect all packets on a particular path. Discard reflects the importance that a packet reach its destination while emission reflects the urgency that a packet reach its destination as quickly as possible.

These attributes are relative to other traffic values for other transmissions. For example, setting all traffic using a particular trunk to the highest emission priority would not accomplish anything since all traffic must wait the same average time before emission to trunk.

Specifying that a path terminate and not reroute

Some applications using HTDS may not tolerate the delays caused by rerouting. To cause a path to terminate instead of rerouting, set the *pathFailureAction* attribute under the *Plc* component to the required value.

Some tips for setting up HTDS on your system

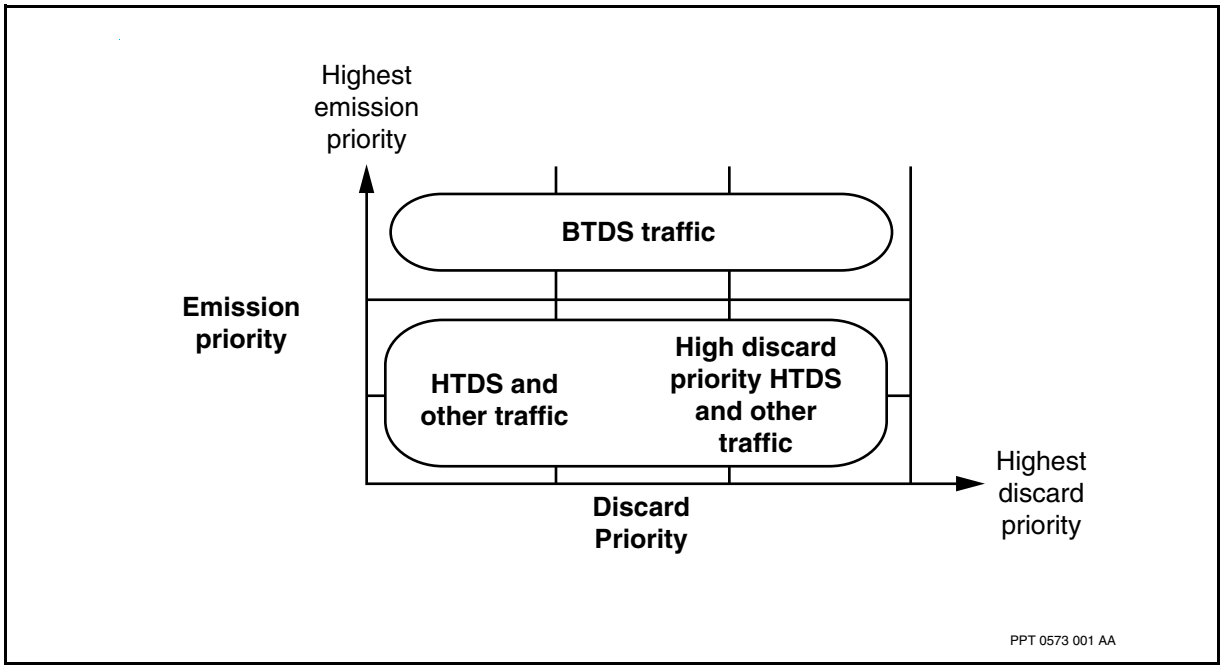
Use this information as a general set of guidelines for using HTDS with other types of traffic. It is not designed to instruct the reader in how to engineer a network.

Emission and discard priority

HTDS uses a strict priority system—higher emission-priority frames get the necessary amount of bandwidth faster. For this reason, too much high-priority traffic will restrict the flow of low-priority traffic. If you are using Bit Transparent Data Service (BTDS) with this service, do not set the amount of frame-cell trunk interrupting mode (BTDS highest priority) traffic at greater than 80% (limit the value for the *maxReservedBwOut* attribute to 80% or less).

In general, HTDS traffic should be set up as shown in figure Emission and discard priority setup for HTDS traffic (page 38). BTDS service information has been included to illustrate how the two services should co-exist.

Emission and discard priority setup for HTDS traffic

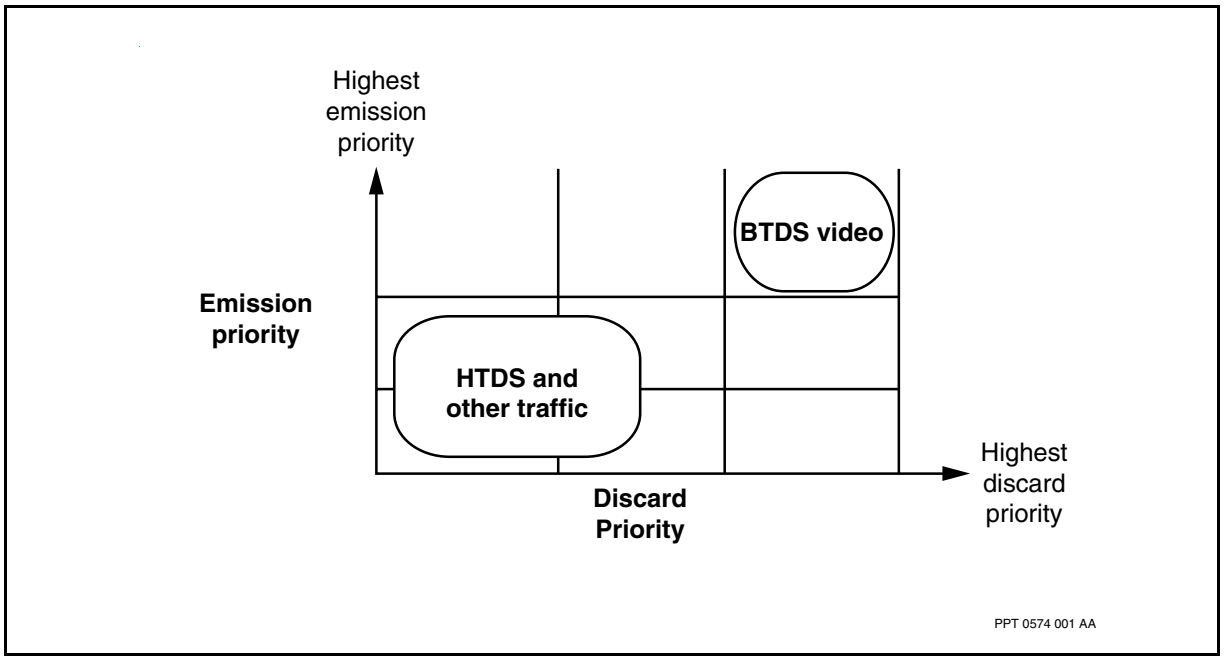


The term highest discard priority means last to be discarded. Highest emission priority means first to be emitted.

Do not forget to account for loads that are not immediately apparent, such as burstiness and control traffic, when determining the amount of HTDS traffic.

When using BTDS with video, HTDS traffic should be set up as shown in figure Emission and discard priority setup for HTDS traffic and BTDS video traffic (page 39).

Emission and discard priority setup for HTDS traffic and BTDS video traffic



Other bandwidth considerations

When considering running HTDS over a pre-existing Nortel Multiservice Switch trunk, determine the normal bandwidth used by the pre-existing traffic. Account for burstiness and control traffic. Estimate the amount of bandwidth needed by the HTDS traffic. If BTDS is also being used, account for the bandwidth needed. Be sure that the combined bandwidth is available.

A large frame size improves node performance and bandwidth utilization but excessively large frames increase delay. Avoid using very large frames for delay-sensitive applications. Generally, a large frame size is beneficial when combined with a high-performance trunk.

Route selection

This section discusses the following topics:

- selecting paths
- restricting traffic
- restricting paths

You can use the criteria which follows to tailor the path that PORS selects to meet your requirements. This can be done during the initial provisioning session or at a later date should you wish to fine-tune the use of your network resources.

Consider the following:

- Reprovisioning causes service interruption. If you reprovision a connection, terminate it and then and re-establish it. The reprovisioning process temporarily stops data flow
- Avoid unnecessary restrictions when provisioning a path. The more restrictions you add, the greater your chance of causing conflicts that will not allow a connection. For example, your restrictions from the security option may require a path that conflicts with the path needed by the general parameters that you have used or that may not support the type of traffic that you want to use. In cases like these, PORS will not be able to set up a connection

Selecting paths

Selecting paths discusses the following topics:

- minimization criteria: cost and delay
- specifying a maximum cost for a path
- specifying a maximum delay for a path

Minimization criteria: cost and delay

PORS can select a path based on either the lowest cost or lowest delay. Cost and delay cannot both be minimized at the same time. Use the *pathAttributeToMinimize* attribute under the *Plc* component to specify cost or delay.

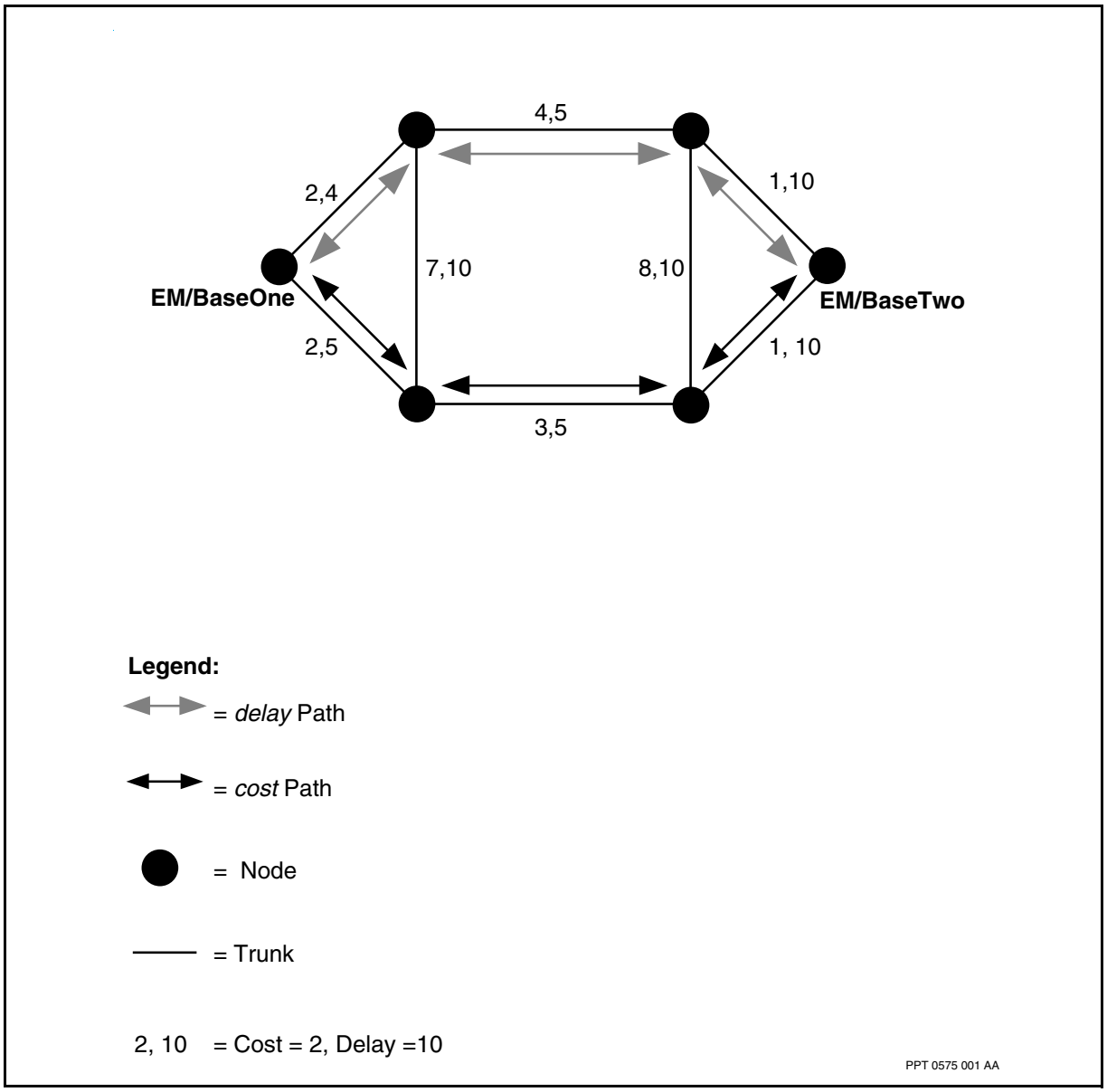
The routing system computes a minimum path from the values that you have assigned to the Nortel Multiservice Switch trunks (cost) or from measured delay values that are associated with each trunk. This is shown in figure Path for cost or delay using the pathAttributeToMinimize attribute (page 41).

To assign a cost to a trunk use the *trunkCost* attribute under the *Trunk* component.

Cost can be an actual dollar value or any parameter that you want to use. If default values are used, cost represents the hop count. Thus the number of hops across the network is minimized.

If you use a parameter for cost that reflects, in some manner, the actual cost of facilities, high-cost facilities will receive less use and reduce the cost of operating the network. This is the recommended method of using this option.

Path for cost or delay using the pathAttributeToMinimize attribute



Specifying a maximum cost for a path

Providers of network services may wish to restrict some parameters for a particular circuit. This can only be done if all the Nortel Multiservice Switch trunks have identical cost.

To specify the maximum total cost value of a path, you can use the *maximumAcceptableCost* attribute under the *Plc* component. Although this value is called cost, you may use it to reflect a variety of considerations, including geographic distance, hop count, or real dollar value.

The sum of the *trunkCost* attribute values of all trunks used in the path will be less than or equal to the value specified by the *maximumAcceptableCost* attribute.

Specifying a maximum delay for a path

Nortel Multiservice Switch trunk delay in PORS is measured for a 512-byte packet in one direction at the time of trunk staging. Over time, this measured delay may change to reflect the updated operating delay but will not affect existing paths unless a trunk restages.

To specify the maximum delay value of a path, use the *maximumAcceptableDelay* attribute under the *Plc* component. The sum of the delay values associated with all trunks used in the path will be less than or equal to the value specified by the *maximumAcceptableDelay* attribute.

This parameter should be used when large delays are unacceptable for the service (for example, for interactive data).

Restricting traffic

Restricting traffic discusses the following topics:

- restricting certain types of traffic to specific Nortel Multiservice Switch trunks
- restricting traffic to certain types of Nortel Multiservice Switch trunks

Restricting certain types of traffic to specific trunks

PORS allows you to specify which types of traffic are carried on a given Nortel Multiservice Switch trunk.

Use the *supportedTrafficTypes* attribute, under the *Trunk* component, to create an individual list of traffic types for each Multiservice Switch trunk in your network (data, voice, and video for example).

Use the *requiredTrafficType* attribute to specify which traffic type is required by the service.

PORS matches a service's *requiredTrafficType* attribute to a trunk's *supportedTrafficTypes* attribute. In other words, the value for a service's *requiredTrafficType* attribute must be included in a trunk's *supportedTrafficTypes* attribute list or the trunk is not selected for the path. For example, if the service's *requiredTrafficType* attribute is provisioned for data, only those trunks with *supportedTrafficTypes* attributes that include data are selected for the path.

As another example, consider two services (HTDS/10 and BTDS/20) running on two trunks over ATM (formerly known as ATM logical trunks): ATM trk/110 for variable bit rate (VBR) data and ATM trk/111 for constant bit rate (CBR) data. To guard against losing frames, provision: HTDS/10 to only use ATM trk/110 (the VBR trunk) and BTDS/20 to only use ATM trk/111 (the CBR trunk).

In this example, video data from the BTDS/20 service is forced onto ATM trk/111 (set up for CBR data) and cannot be routed onto ATM trk/110 (set up for VBR data).

Restricting traffic to certain types of trunks

You may want to create an indicator of the type of Nortel Multiservice Switch trunk that various traffic types use. Terrestrial or satellite links are examples of Multiservice Switch trunking facilities. The *trunkType* attribute, under the *Trunk PathAdministrator* component, allows you to do this for up to eight different types of Multiservice Switch trunks.

The *permittedTrunkTypes* attribute under the *Plc* component allows a set of possible trunk types to be specified for a route. Only trunks with *trunkType* attributes that are found in the *permittedTrunkTypes* attribute list are used to create the path.

Restricting paths

Restricting paths discusses the following topics:

- security
- defining general parameters to restrict paths
- specifying a path manually

Security

PORS allows you to define varying security levels for the Nortel Multiservice Switch trunks of the network. This option could be used to prevent sensitive data from traveling over certain Multiservice Switch trunks, for example.

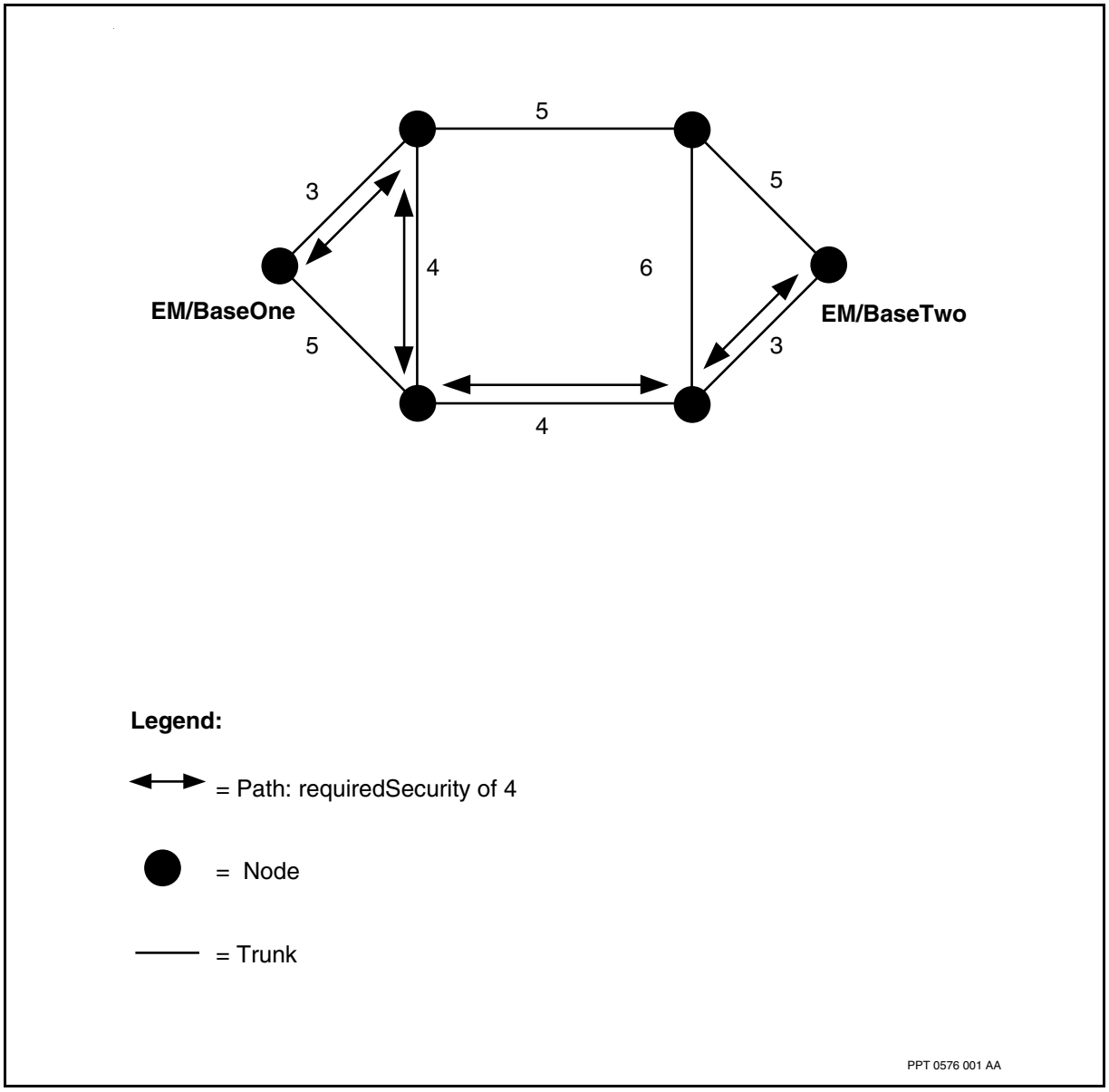
PORS has an option that allows you to specify the minimum security level of a path. To do this, provision a security value for the trunks in your Nortel Multiservice Switch 7400 network using the *trunkSecurity* attribute under the *Trunk PathAdministrator* component. When you provision the connection, enter a value for the *requiredSecurity* attribute under the *Plc* component.

The connection will only use trunks that have been assigned security values of an equal or higher level than that of the connection. This is illustrated in the example in figure Path determined using a value of 4 for the requiredSecurity attribute (page 44). A lower number always represents a higher security level.

The default value for security is mid-range so that the network administrator can add security with minimal provisioning.

Over-use of this option can reduce its usefulness. This option can also reduce the number of recovery paths available to high security routes should an outage occur.

Path determined using a value of 4 for the requiredSecurity attribute



Defining general parameters to restrict paths

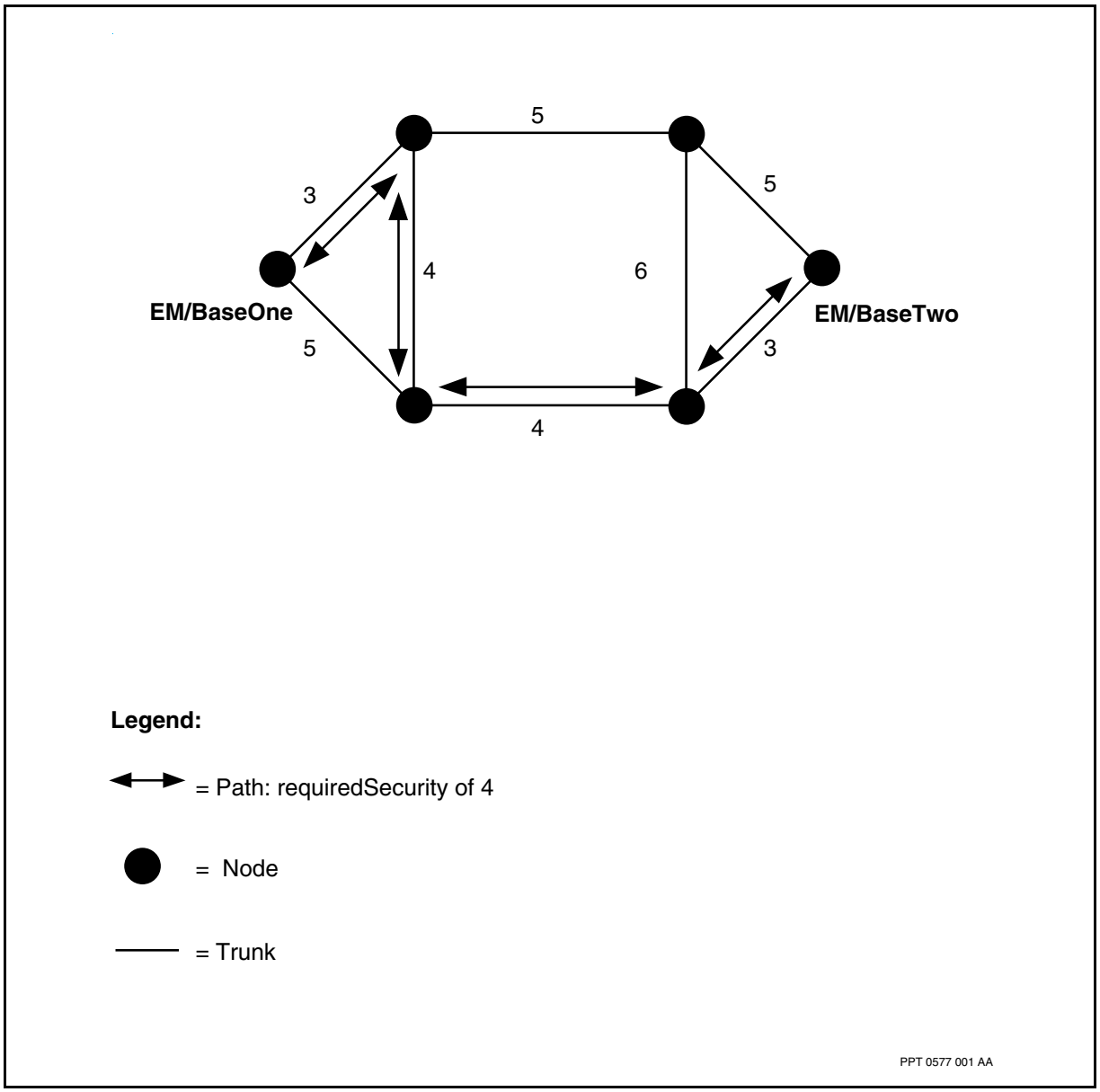
It may be convenient to be able to restrict certain classes of paths to certain Nortel Multiservice Switch trunks. Most of the commonly used qualifiers are represented in security and traffic type. This is an additional option, to be used for any function that you deem appropriate.

PORS allows you to restrict certain paths to certain trunks. This is done in a similar manner to the way that security is provisioned. Values are assigned to various trunks in the network using the *customerParameter* attribute under the *Trunk PathAdministrator* component. When the *Plc* component is provisioned, it can be assigned a value using the *requiredCustomerParameter* attribute.

PORS will assign the path to trunks that have an equal or lower number associated with them. This is illustrated in the example shown in figure Path using a value of 4 for the requiredCustomerParameter attribute (page 46).

All restrictions are applied simultaneously during route selection. Over-restricting trunks and the PLC may result in no route being selected, where different trunks would be rejected for different reasons.

Path using a value of 4 for the requiredCustomerParameter attribute



Specifying a path manually

PORS is designed to select an appropriate route automatically. In an exceptional case, however, you may wish to define the set of Nortel Multiservice Switch trunks that are to be used.

The route can be defined at both end points. The two routes do not have to use the same set of trunks. If different routes are defined at each end, PORS does not guarantee which one will be used.

Defining different routes at both end points has an advantage. This simple provisioning provides a backup route for manual path in case of a failure impacting the route in use. Different manual routes enhance robustness of Voice Transport.

If you want to override the automatic selection of a path and specify the trunks manually, use the *manualPath* attribute under the *Plc* component. Enter the outbound sequence of trunk component names for the path that you want.

The path must satisfy the characteristics specified in the other attributes under the *Plc* component, including bandwidth requirements.

The table Path and trunk attributes under the PermanentLogicalConnection component (page 47), lists the path and trunk attributes under the *PermanentLogicalConnection* component.

Path and trunk attributes under the PermanentLogicalConnection component

Path requirement attribute	Trunk attribute	Trunk is candidate for route if...
pathAttributeToMinimize (value = cost or delay)	trunkCost, trunkDelay	The value for the *trunkCost* attribute is less than the value for the *maximumAcceptableCost* attribute. The value for the *trunkDelay* attribute is less than the value for the maximumAcceptableDelay attribute.
maximumAcceptableCost (value = number)	trunkCost (value = number)	
maximumAcceptableDelay (value = number of milliseconds)	trunk delay metric (Metric in milliseconds: not provisionable)	
requiredTrafficType value = one of eight traffic types (0 to 7) (voice, data, video, 3, 4, 5, 6, or 7)	*supportedTrafficTypes* (value = list of up to eight traffic types)	The value for the *requiredTrafficType* attribute is included in the values for the *supportedTrafficTypes* attribute.
permittedTrunkTypes (value = list of up to eight trunk types)	*trunkType* value = one of 0 to 7 (terr, sat, 2, 3, 4, 5, 6, or 7)	The value for the *trunkType* attribute is included in the values for the *permittedTrunkTypes* attribute.
requiredSecurity (value = number)	*trunkSecurity* (value = number)	The value for the *trunkSecurity* attribute is less than or equal to the value for the *requiredSecurity* attribute.
requiredCustomerParameter (value = number)	*trunkCustomerParameter* (value = number)	The value for the *trunkCustomerParameter* attribute is less than or equal to the value for the *requiredCustomerParameter* attribute.

(1 of 2)

Path and trunk attributes under the PermanentLogicalConnection component (continued)

Path requirement attribute	Trunk attribute	Trunk is candidate for route if...
manualPath (value = list of trunk component IDs)	*trunk component name* (for example: EM/NODER9 trunk/202)	The *Trunk* component name is one of *manualPath*
pathAttributeToMinimize (value = cost or delay)	trunkCost, trunkDelay	The value for the *trunkCost* attribute is less than the value for the *maximumAcceptableCost* attribute. The value for the *trunkDelay* attribute is less than the value for the maximumAcceptableDelay attribute.
(2 of 2)		

HDLC Transparent Data Service engineering guidelines

Use this chapter to learn about engineering aspects that affect the HDLC Transparent Data Service and illustrates their effect on the operating characteristics of this service.

Navigation

Services required by the HDLC Transparent Data Service

The HDLC Transparent Data Service is supported by the Nortel Multiservice Switch Path Oriented Routing System (PORS). PORS provides the underlying routing system for the HDLC Transparent Data Service in a Nortel Multiservice Switch 7400 network. PORS ensures that HDLC frames converted into packets encounter minimal network delay and that packet ordering is preserved across the network. Refer to Establishing connections (page 30) for additional information about PORS.

System capabilities

The HDLC Transparent Data Service system supports

- a maximum packet size of 4096 bytes
- octet or non-octet aligned data (support of non-octet aligned data on HDLC Transparent Data Service reduces the maximum throughput by 30%)
- up to eight HDLC Transparent Data Service interfaces using V.35 function processor (FP) physical connections with speeds from 9.6 kbit/s to 2 Mbit/s for each interface

- up to eight HDLC Transparent Data Service interfaces using V.11 FP physical connections with speeds from 9.6 kbit/s to 2 Mbit/s for each interface

  There is a maximum combined throughput for the eight interfaces of the V.11 FP. See NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description* for more information.

- up to four HDLC Transparent Data Service interfaces using DS1, 8pDS1 or E1 FP physical connections in single-link mode
- In single-link mode, one primary rate interface can have only one channel associated with it. The channel can have from 1 to 24 timeslots on a DS1 port and 1 to 31 timeslots on an E1 port.
- up to eight HDLC Transparent Data Service interfaces using DS1, 8pDS1 or E1 physical connections in fractional mode (n*64)

  In fractional mode, one primary rate interface can have one to four channels associated with it. This may be done with ports 1 and 3. For a description of the DS1, 8pDS1 and E1 FPs, see NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description*.

- transparent transfer of modem status leads using either V.35 FP or V.11 FP
- a single HDLC Transparent Data Service interface using a HSSI FP physical connection (he HSSI FP cannot be used for network clock synchronization)

Service interconnection requirements

Source and destination FPs (terminating FPs or access point FPs) can be of the same type (V.11 to V.11 for example) or different (V.35 to E1). The following is a list of requirements for all possible interconnections:

- V.11 to V.11 or V.35 to V.35

  The S (V.11) and REST (V.35) clock values must be supported by V.11 or V.35. See NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description* for values of generated clock rates. Examples of clock rates are: 9.6, 19.2, 32, 48, 56, 64, 112, 128, 168, 192, 224, 256, 320, 336, 384, 448, 512, 640, 672, 768, 960, 1280, 1344, 1536, and 1920 kHz.

  Modem status leads can be terminated or sent transparently through the network.

- DS1 or 8pDS1 to DS1 or 8pDS1

  In fractional (multi-channel) mode, each port supports up to four fractions:

  — Each channel can contain 1 to 24 timeslots in increments of one timeslot at a rate of 56 kbit/s (using bit 7 stuffing) or 64 kbit/s for each timeslot.

— The total number of timeslots for each port is 24.

— In single-channel mode, each port supports a single fraction: the channel can contain 1 to 24 timeslots in increments of one timeslot at a rate of 56 kbit/s (using bit 7 stuffing) or 64 kbit/s for each timeslot.

- E1 to E1

  In fractional (multi-channel) mode, each port supports up to four fractions:

  — Each channel can contain 1 to 31 timeslots in increments of one timeslot at a rate of 64 kbit/s for each timeslot.

  — The total number of timeslots for each port is 31.

  — In single-channel mode, each port supports a single fraction: the channel can contain 1 to 31 timeslots in increments of one timeslot at a rate of 64 kbit/s for each timeslot.

- V.11 or V.35 to DS1, 8pDS1 or E1

  — Line status leads must be terminated locally (V.11 and V.35).

  — Channel selections are 1, 2, 3, 4, 6, 8, 12, or 24 for DS1 interfaces operating at a timeslot rate of 56 kbit/s.

  — Channel selections are 1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 15, 20, 21, or 24 for DS1 interfaces operating at a timeslot rate of 64 kbit/s.

  — Channel selections are 1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 15, 20, 21, 24, or 31 for E1 operating at a timeslot rate of 64 kbit/s.

- V.11 to V.35

  — Line status leads must be terminated locally.

- DS3 to DS3
- DS3 to DS1 or 8pDS1
- HSSI to HSSI
- HSSI to DS1 or 8pDS1
- HSSI to DS3

Nortel Multiservice Switch 7400

Operations: HDLC Transparent Data Service

Publication: NN10600-770
Document status: Standard
Document issue: 7.1S1
Document date: October 2005
Product release: PCR7.1 and up
Job function: Operations
Type: NTP
Language type: U.S. English