NORTEL NETWORKS

Nortel Networks Multiservice Switch 7400

Operations: HDLC Transparent Data Service

NN10600-770

Nortel Networks Multiservice Switch 7400

Operations: HDLC Transparent Data Service

Publication: NN10600-770
Document status: Standard
Document version: 6.1S1
Document date: August 2004

Publication history

August 2004

6.1S1 Standard
General availability. Contains information on Nortel Networks Multiservice Switch 7400 for the PCR6.1 release.

Contents

List of figures

List of tables

About this document

The NN10600-770 *Nortel Networks Multiservice Switch 7400 Operations: HDLC Transparent Data Service* describes Nortel Networks Multiservice Switch 7400 HDLC Transparent Data Service (HTDS).

The following topics are discussed in this section:

Who should read this document and why

This document is for persons with the following HTDS responsibilities:

- planning
- engineering
- installing and configuring
- provisioning, operating, maintaining, and troubleshooting

What you need to know

This document assumes that you have an understanding of Nortel Networks Multiservice Switch 7400 network architecture.

How this document is organized

NN10600-770 *Nortel Networks Multiservice Switch 7400 Operations: HDLC Transparent Data Service* contains the following sections:

- "HDLC Transparent Data Service configuration" (page 19) provides detailed procedures for provisioning the HDLC Transparent Data Service in your network.
- "HDLC Transparent Data Service fundamentals" (page 33) is an overview of the HDLC Transparent Data Service, Path Oriented Routing System (PORS).
- "HDLC Transparent Data Service engineering guidelines" (page 61) outlines the aspects of HDLC Transparent Data Service engineering that must be addressed to ensure that service operates efficiently.
- "Component monitoring" (page 65) describes common operational mode commands, and how to use them for monitoring Bit transparent data components and attributes.
- "Troubleshooting" (page 75) outlines how to troubleshoot the HTDS service if any problems are encountered after installation.

What's new in this document

There were no new features added to this document.

Other changes made to this document include the following:

- The terms Passport and PVG have been rebranded in conjunction with the new Nortel Networks' brand simplified naming format. Passport is now referred to as the Nortel Networks Multiservice Switch, and PVG is now Media Gateway 7480/15000. For more information on the product rebranding, refer to NN10600-000 *Nortel Networks Multiservice Switch 7400/15000/20000 What's New in PCR6.1*.

Text conventions

This document uses the following text conventions:

- `nonproportional spaced plain type`

  Nonproportional spaced plain type represents system generated text or text that appears on your screen.

- **nonproportional spaced bold type**

  Nonproportional spaced bold type represents words that you should type or that you should select on the screen.

- *italics*

  Statements that appear in italics in a procedure explain the results of a particular step and appear immediately following the step.

  Words that appear in italics in text are for naming.

- [optional_parameter]

  Words in square brackets represent optional parameters. The command can be entered with or without the words in the square brackets.

- <general_term>

  Words in angle brackets represent variables which are to be replaced with specific values.

- UPPERCASE, lowercase

  Uppercase and lowercase letters that appear in UNIX commands and parameters must be matched exactly. The system matches upper and lowercase characters differently.

- Nortel Networks Multiservice Switch commands are not case-sensitive and do not have to match commands and parameters exactly as shown in this document, with the exception of string options values (for example, file and directory names) and string attribute values.

- |

  This symbol separates items from which you may select one; for example, ON|OFF indicates that you may specify ON or OFF. If you do not make a choice, a default ON is assumed.

- ...

  Three dots in a command indicate that the parameter may be repeated more than once in succession.

The term absolute pathname refers to the full specification of a path starting from the root directory. Absolute pathnames always begin with the slash (/) symbol. A relative pathname takes the current directory as its starting point, and starts with any alphanumeric character (other than /).

Related documents

For the complete list of documents contained in the Nortel Networks Multiservice Switch documentation library, see NN10600-001 *Nortel Networks Multiservice Switch 7400/15000/20000 Basics: Customer Documentation*. Refer to the following documents for information on installing and operating HTDS in your network:

- NN10600-030 *Nortel Networks Multiservice Switch 7400/15000/20000 Overview*
- NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*
- NN10600-175 *Nortel Networks Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*
- NN10600-551 *Nortel Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference*
- NN10600-270 *Nortel Networks Multiservice Switch 7400/15000/20000 Software Installation*
- NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*
- NN10600-420 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Trunking*
- NN10600-435 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*

How to get more help

For information on training, problem reporting, and technical support, see the "Nortel Networks support services" section in the product overview document.

Chapter 1
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

- You will need information contained in NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference* to successfully provision the HDLC Transparent Data Service and Path Oriented Routing System.
- Ensure that 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 Networks Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System* for information to provision with PORS.
- To specify the HTDS application, add vtds or hdlcTransparent to the *featureList* attribute using the following command:

```
set software lpt/<instance> featureList <feature>
```

where:
<instance> is the descriptive name given to this *lpt* component instance.
<feature> can be 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.

- Verify that the logical processor type for Trunks has path-oriented software in its feature list. For example, you can use the command:

```
set software lpt/trunks featureList porsTrunks
```

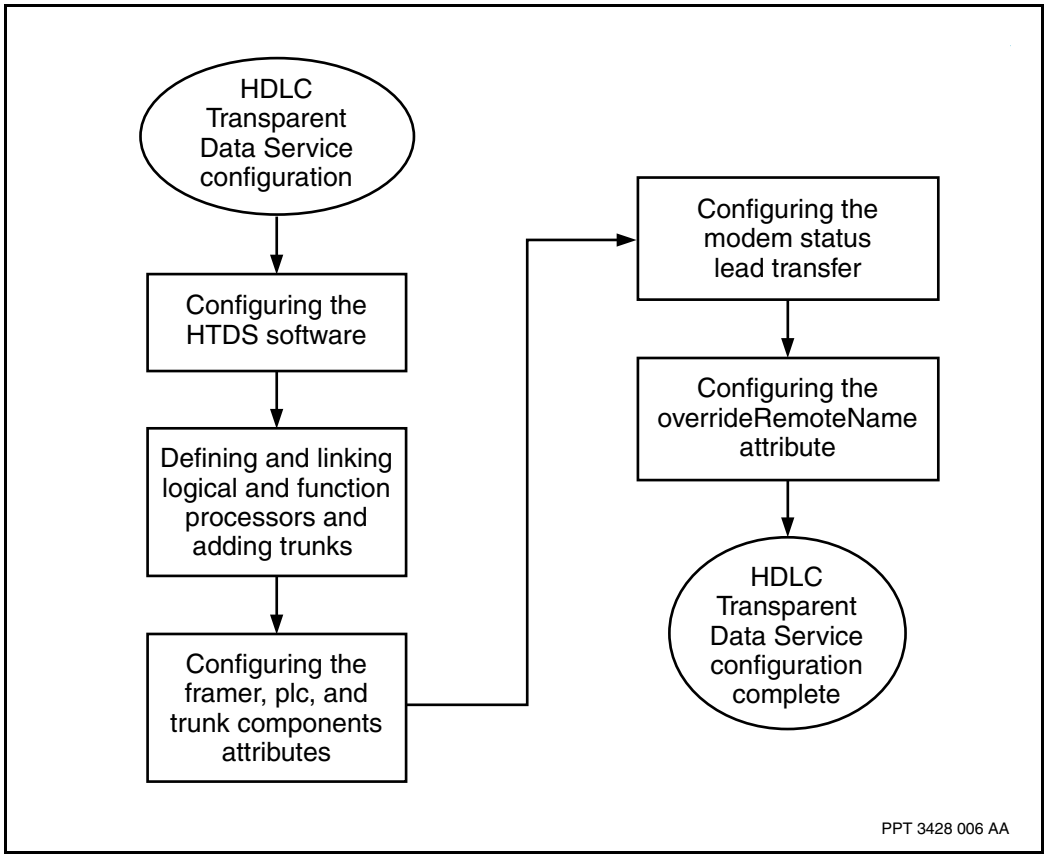
All nodes in the network that are candidates for HTDS traffic must be running the latest software and must have the *Trunk PathAdministrator* component added to the view.

- NN10600-270 *Nortel Networks Multiservice Switch 7400/15000/20000 Software Installation* describes how to install the software for this service.
- For information on using the Nortel Networks Multiservice Switch command-line interface, see NN10600-050 *Nortel Networks Multiservice Switch 7400/15000/20000 Command Reference*.

HDLC Transparent Data Service configuration task

"HDLC Transparent Data Service configuration task flow" (page 21) shows you the sequence of tasks and procedures you perform to configure the HDLC Transparent Data Service. To link to any task or procedure, go to "Task navigation" (page 21).

Figure 1
HDLC Transparent Data Service configuration task flow



Task navigation

- "Configuring the HTDS software" (page 22)
- "Defining and linking logical processors and adding trunks" (page 23)
- "Configuring the attributes of the Framer, Plc, and Trunk components" (page 26)
- "Configuring modem status lead transfer" (page 29)
- "Configuring the overrideRemoteName attribute" (page 31)

Configuring the HTDS software

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

Procedure steps

1. Define an instance of the *lpt* attribute.

   ```
   add software lpt/<name>
   ```

2. Add the HTDS application.

   ```
   set software lpt/<name> featureList <feature>
   ```

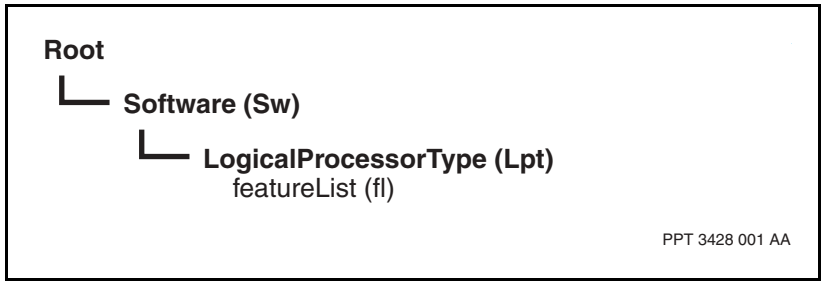
3. Repeat step 1 and step 2 to configure HTDS software on the remote end node.

Variable definitions

Variable	Definition
<feature>	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>	Up to 25 ASCII characters. To help simplify provisioning, use a descriptive word when naming an *Lpt* component, such as TDS.

Procedure job aid

Figure 2
Configuring the 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

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

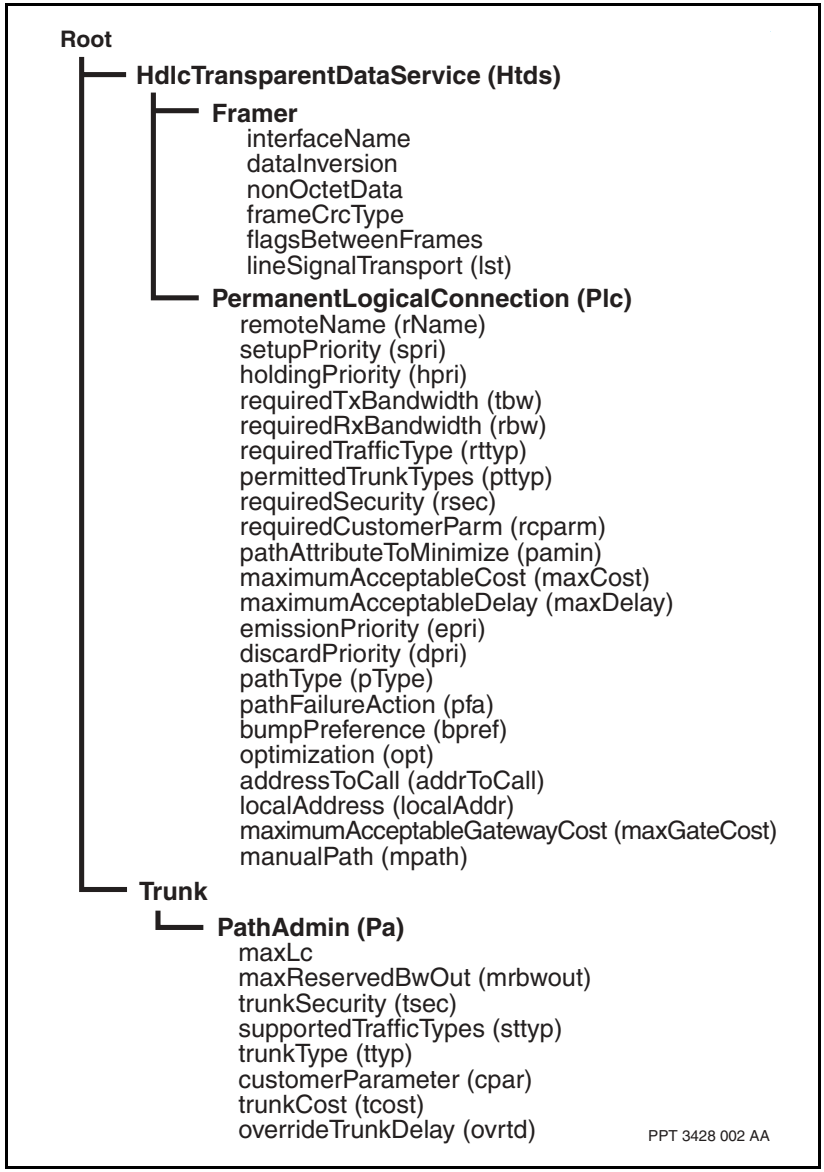
9. Repeat step 1 through step 8 to define and link logical processors and to add trunks to the other end of the connection.

Variable definitions

Variable	Definition
<htds_number>	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>	The logical processor value that you assigned when you provisioned the card.
<name>	The name you assigned to the HTDS software (for example, TDS).
<port_number>	The port number that you assigned when you provisioned the port.
<port_type>	X21 (for a V.11 port) or V35 (for a V.35 port).
<remote_name>	The PLC's remote end-point's address; 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>	The instance number of the Trunk component. See NN10600-420 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Trunking* for more information about adding a Nortel Networks Multiservice Switch trunk.

Procedure job aid

Figure 3
Defining and linking logical processors and adding trunks component hierarchy



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

You can 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 Networks Multiservice Switch 7400/15000/20000 Component Reference*.

Procedure steps

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>

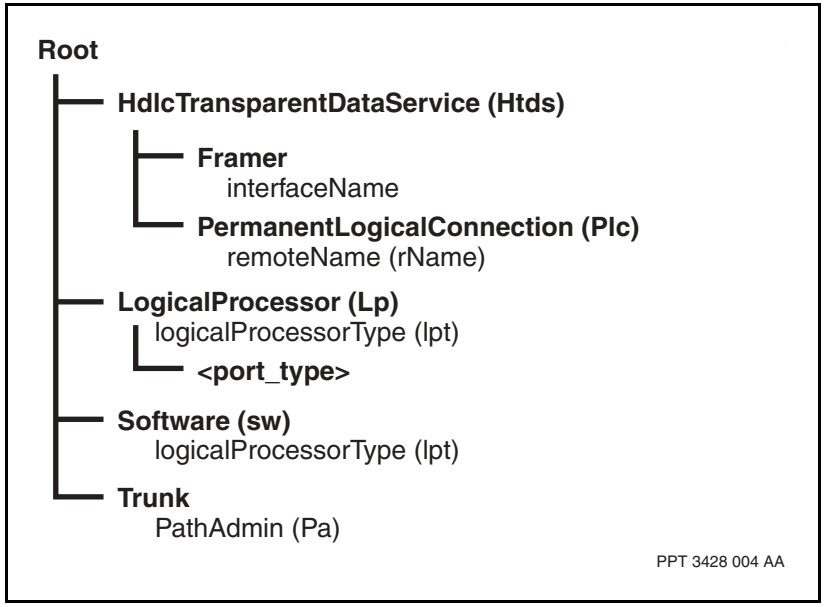
4. Repeat step 1 through step 3 to configure the attributes of the Framer, Plc, and Trunk components at the other end of the connection.

Variable definitions

Variable	Definition
<framer_attribute>	One of the provisionable attributes of the Framer component.
<framer_attribute_value>	The value of the Framer component's provisionable attribute.
<htds_number>	The instance number of the Htds component.
<plc_attribute>	One of the provisionable attributes of the Plc component.
<plc_attribute_value>	The 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>	One of the provisionable attributes of the Trunk component.
<trunk_attribute_value>	The value of the Trunk component's provisionable attribute.
<trunk_number>	The instance number of the trunk.

Procedure job aid

Figure 4
Configuring the attributes of the Framer, Plc, and Trunk components component hierarchy



Configuring modem status lead transfer

Configure modem status lead transfer to send status lead changes transparently through the network. Set the readyLineState and dataTransferLineState attributes to ignore the incoming status leads and set the lineSignalTransport attribute to on. Set one end of the Nortel Networks Multiservice Switch subnet set to DTE and the other end to DCE.

Procedure steps

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.

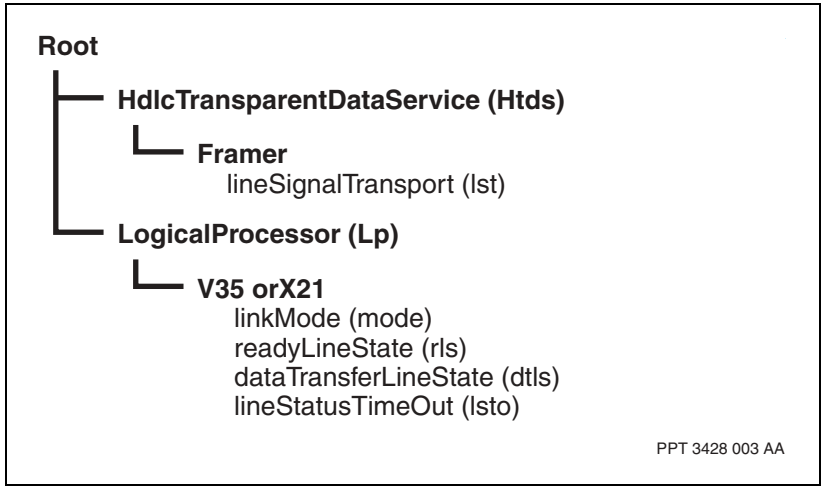
   ```
   set htds/<htds_number> framer lineSignalTransport on
   ```

Variable definitions

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

Procedure job aid

Figure 5
Configuring the modem status lead transfer component hierarchy



Configuring the overrideRemoteName attribute

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

Procedure steps

1 Set the *overrideRemoteName* attribute.

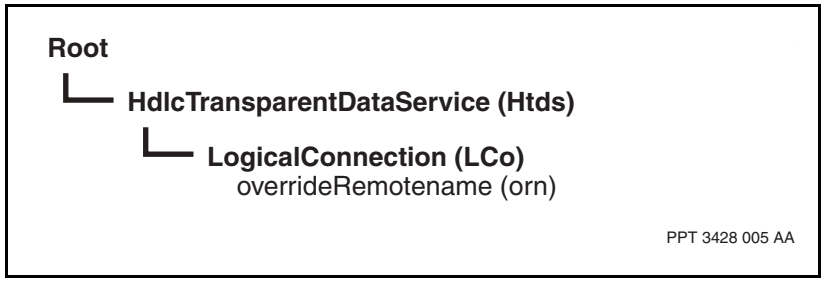
```
set htds/<htds_number> lc overrideRemoteName
<override_name>
```

Variable definitions

Variable	Definition
<htds_number>	The instance number of the Htds component.
<override_name>	An override name for the PLC's remote end-point's address.

Procedure job aid

Figure 6
Configuring the overrideRemoteName attribute component hierarchy



Chapter 2
HDLC Transparent Data Service fundamentals

This chapter details the Nortel Networks Multiservice Switch HDLC Transparent Data Service. See the following topics:

The terms service or HDLC service, as used in this guide, refer to the Nortel Networks Multiservice Switch 7440 HDLC Transparent Data Service.

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 Networks 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

  Note: 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 Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference*.

For more information see

- NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*
- NN10600-175 *Nortel Networks Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*
- NN10600-551 *Nortel Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference*
- NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*
- NN10600-500 *Nortel Networks 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 38) 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.

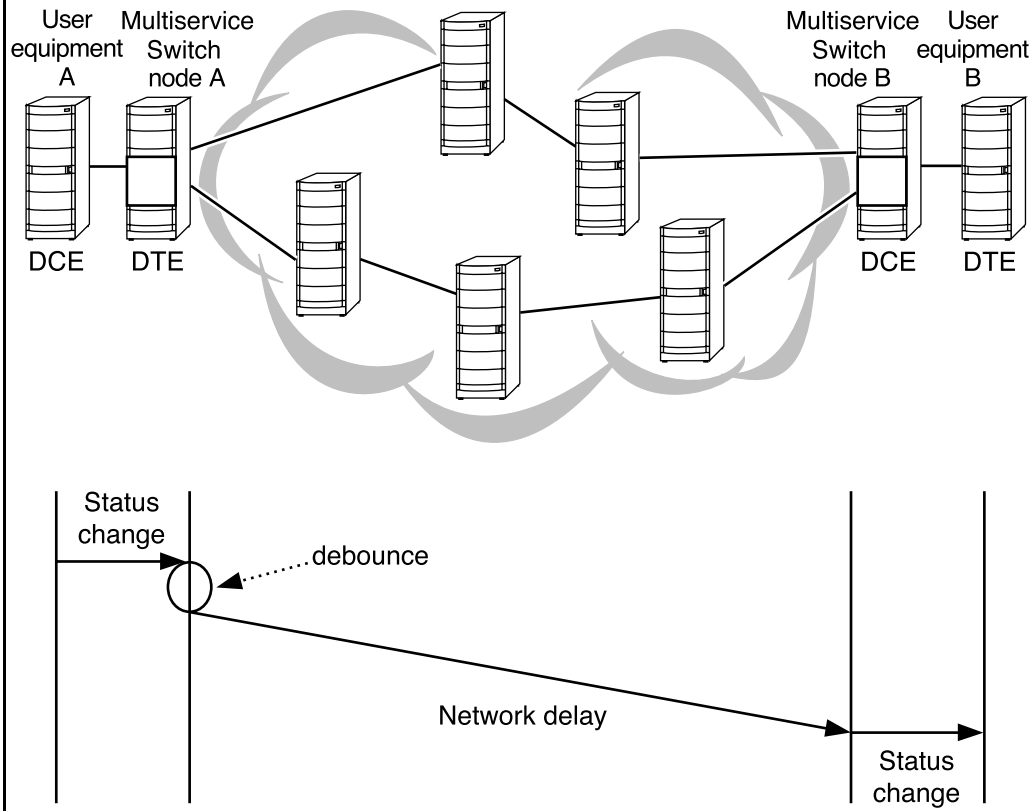
The modem status lead feature is used when the following conditions are met:

- 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 29).

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 of figure "Modem status lead transfer in a Multiservice Switch network" (page 38) 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 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.

Figure 7
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).

MSS 0758 001 AA

Establishing connections

Figure “Path and path description” (page 41) 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 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 contain information about establishing connections across a network:

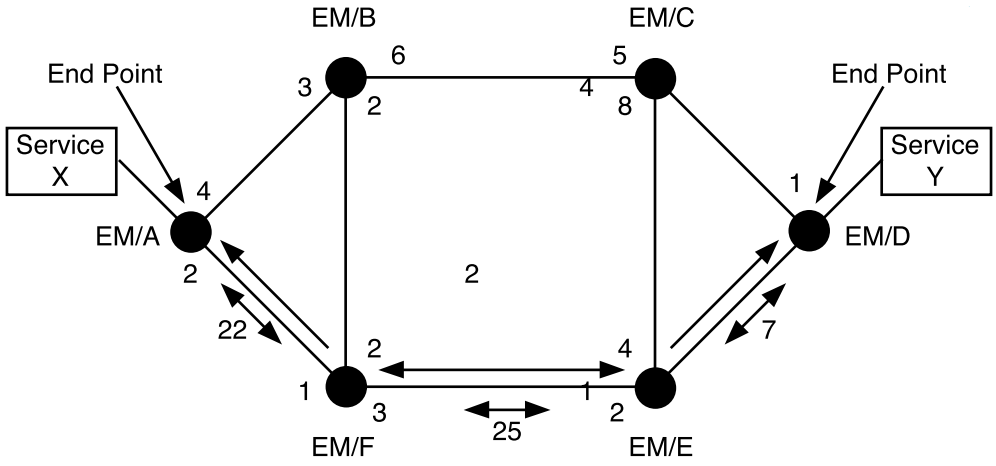
Establishing a path

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 19). 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 Networks Multiservice Switch 7400/15000/20000 Component Reference*.

Note: 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.

Figure 8
Path and path description



Legend:

2 = Trunk instance

25 = LCh - Logical Channel Number

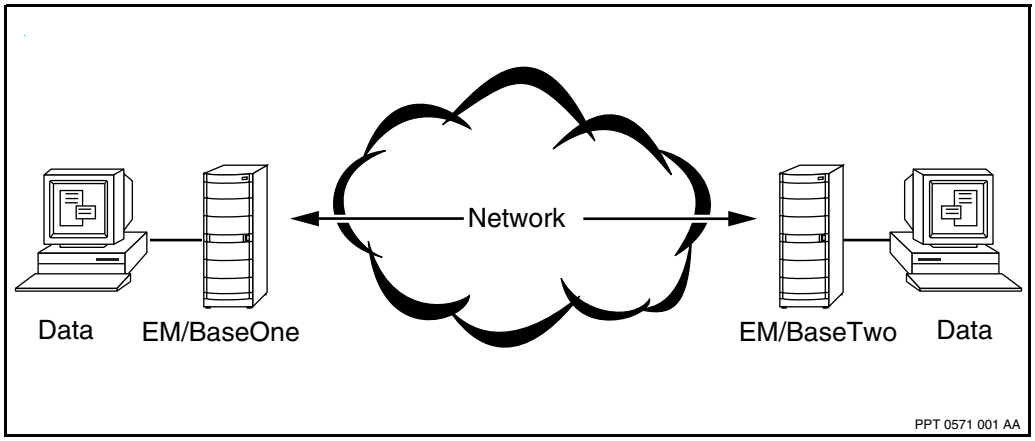
= Node

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

Figure 9
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 (PLC) 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 49).

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 Networks 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 Networks 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 Networks Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*.

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

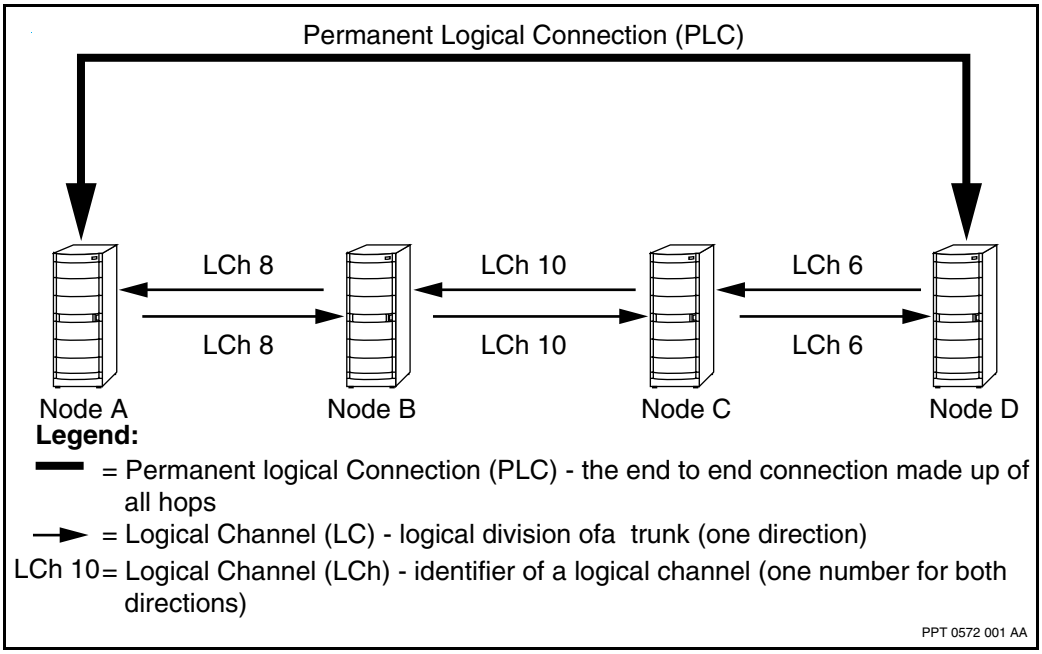
Recovering from path establishment failure

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 Networks 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.

Figure 10
PLCs, LCs, and LChs



Multiservice Switch trunk bandwidth allocation

This section lists provisionable parameters, provided by a Nortel Networks 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 Networks 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 Networks 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 Networks 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:

Note: This is an example only. The values listed below are not the default settings.

Table 1
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 Networks 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 Networks 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.

> ***Note:*** 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

This discussion is meant to be 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.

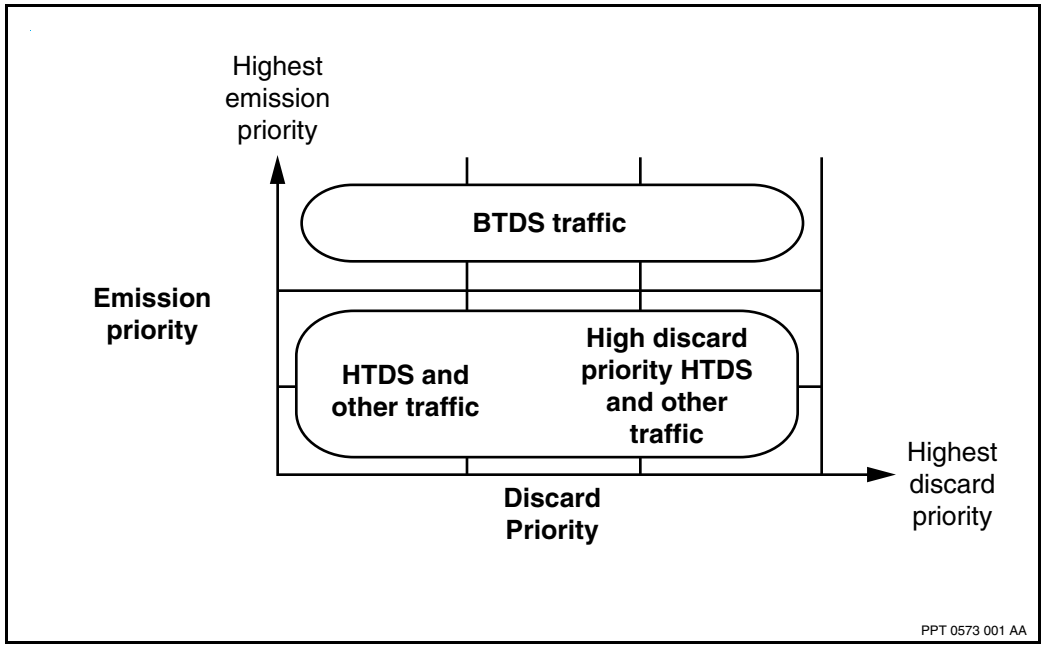
Emission and discard priority

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 the following emission versus discard priority diagram. BTDS service information has been included to illustrate how the two services should co-exist.

Figure 11
Emission and discard priority setup for HTDS traffic

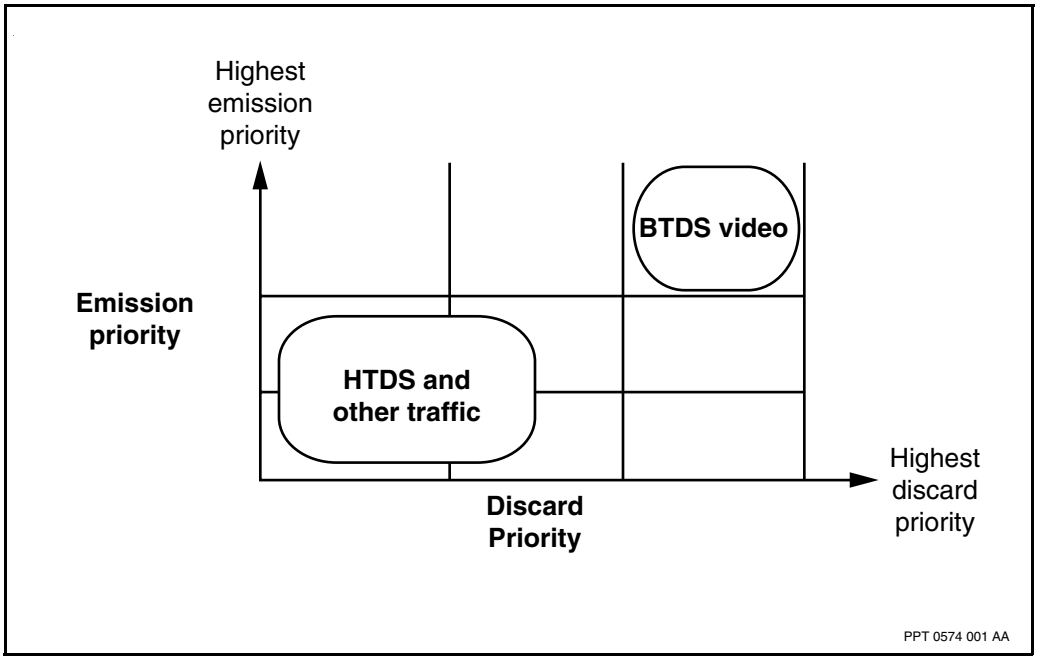


Note: The term highest discard priority means last to be discarded. Highest emission priority means first to be emitted.

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 the following emission versus discard priority diagram.

Figure 12
Emission and discard priority setup for HTDS traffic and BTDS video traffic



Other bandwidth considerations

When considering running HTDS over a pre-existing Nortel Networks 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 following criteria 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.

Note 1: Reprovisioning causes service interruption. If you reprovision a connection, you terminate and re-establish it. The reprovisioning process temporarily stops data flow.

Note 2: 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.

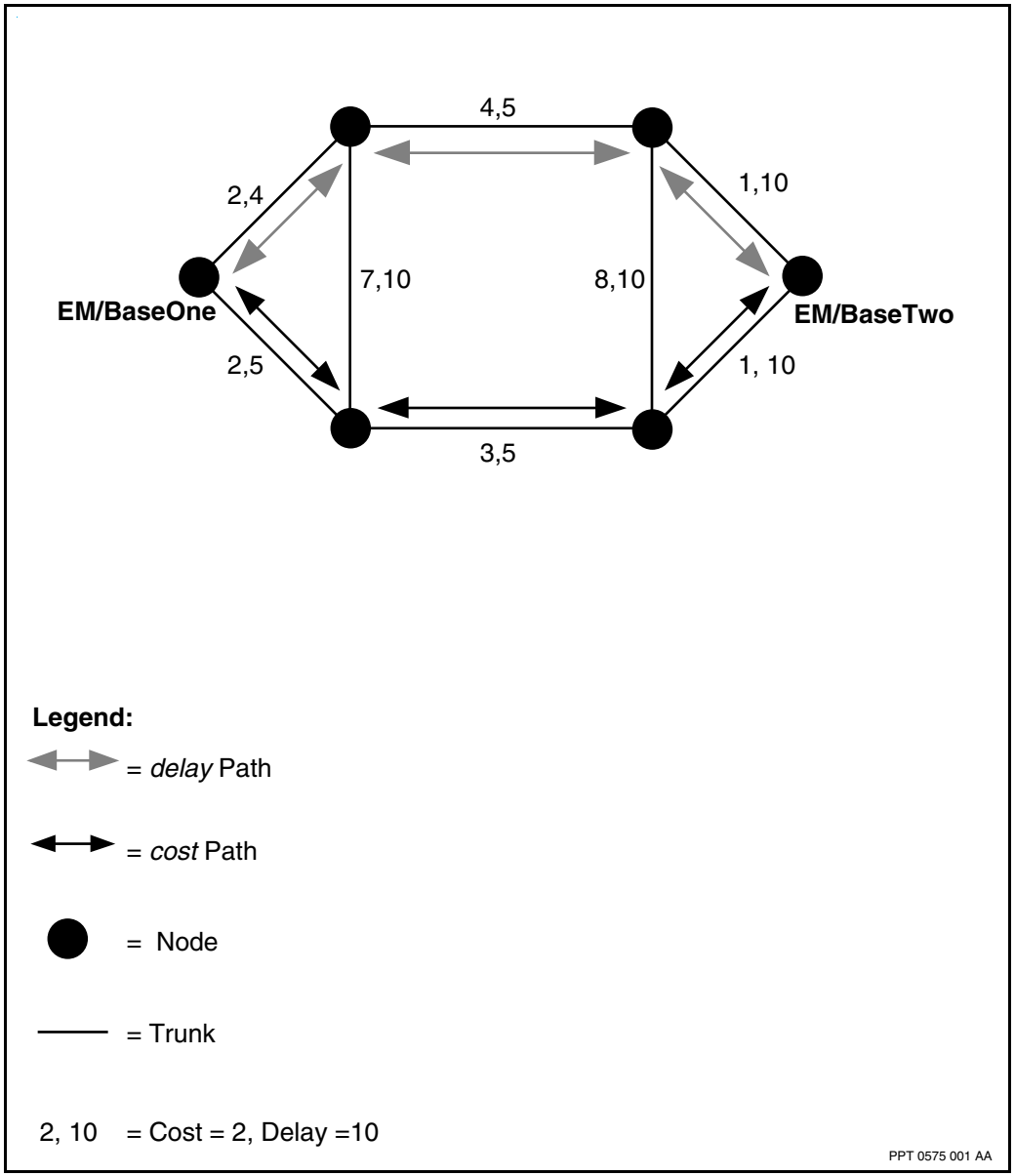
The routing system computes a minimum path from the values that you have assigned to the Nortel Networks Multiservice Switch trunks (cost) or from measured delay values that are associated with each trunk. This is shown in figure “Path for cost or delay using the trunkAttributeToMinimize attribute” (page 52).

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.

Figure 13
Path for cost or delay using the trunkAttributeToMinimize 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 Networks 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 Networks 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.

> ***Note:*** 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 Networks Multiservice Switch trunks
- restricting traffic to certain types of Nortel Networks 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 Networks 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). The provisioning required to accomplish this would include the following steps:

```
set htds/10 plc requiredTrafficType data

set btds/20 plc requiredTrafficType video

set trk/110 pa supportedTrafficTypes data ~video

set trk/111 pa supportedTrafficTypes video ~data
```

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 Networks 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 Networks Multiservice Switch trunks of the network. This option could be used to prevent sensitive data from traveling over certain Multiservice Switch trunks, for example.

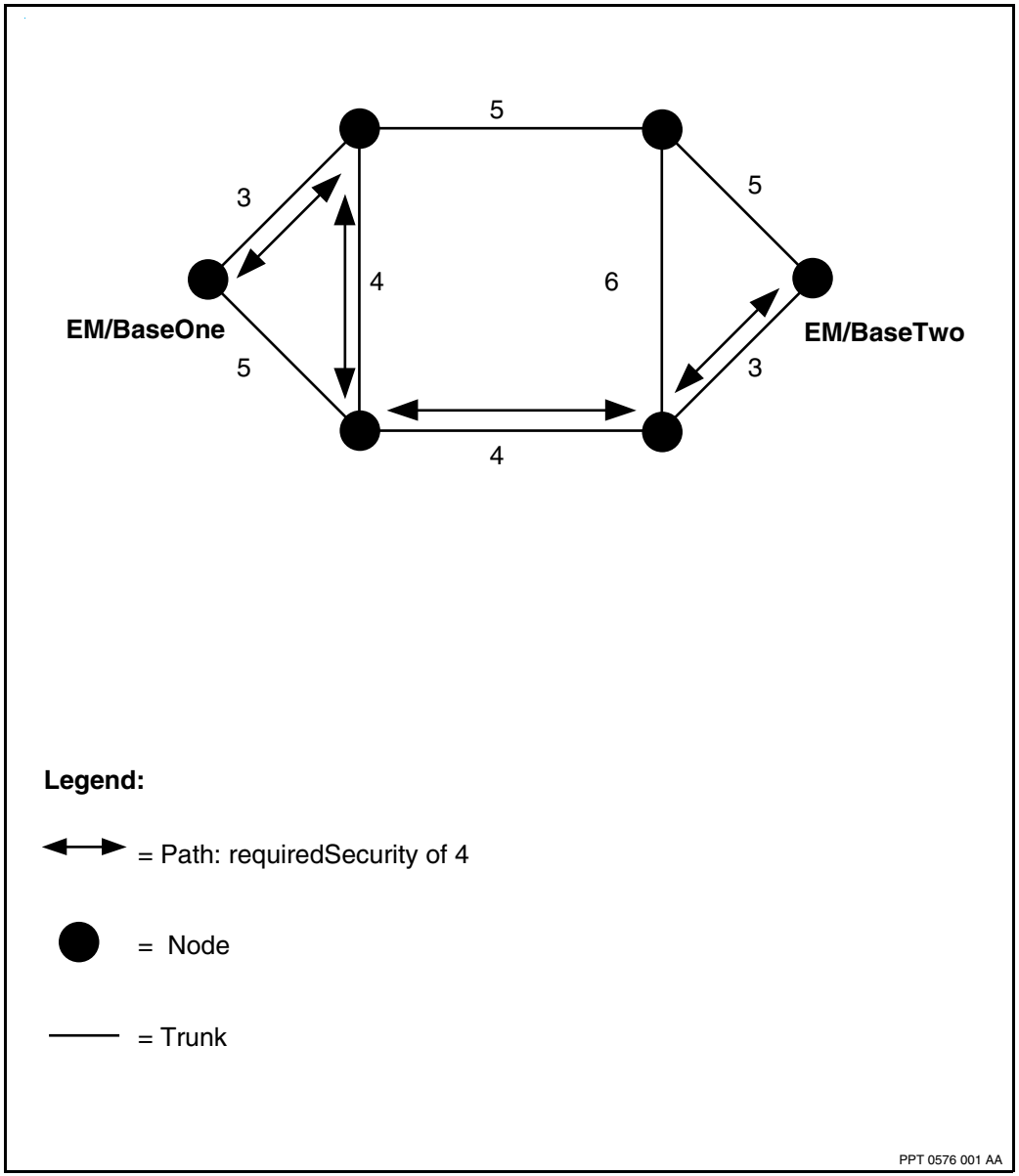
PORS has an option that allows you to specify the minimum security level of a path. To do this, provision a security value for the trunks in your Nortel Networks Multiservice Switch 7400 network using the *trunkSecurity* attribute under the *Trunk PathAdministrator* component. When you provision the 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 56). 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.

Note: 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.

Figure 14
Path determined using a value of 4 for the requiredSecurity attribute



Defining general parameters to restrict paths

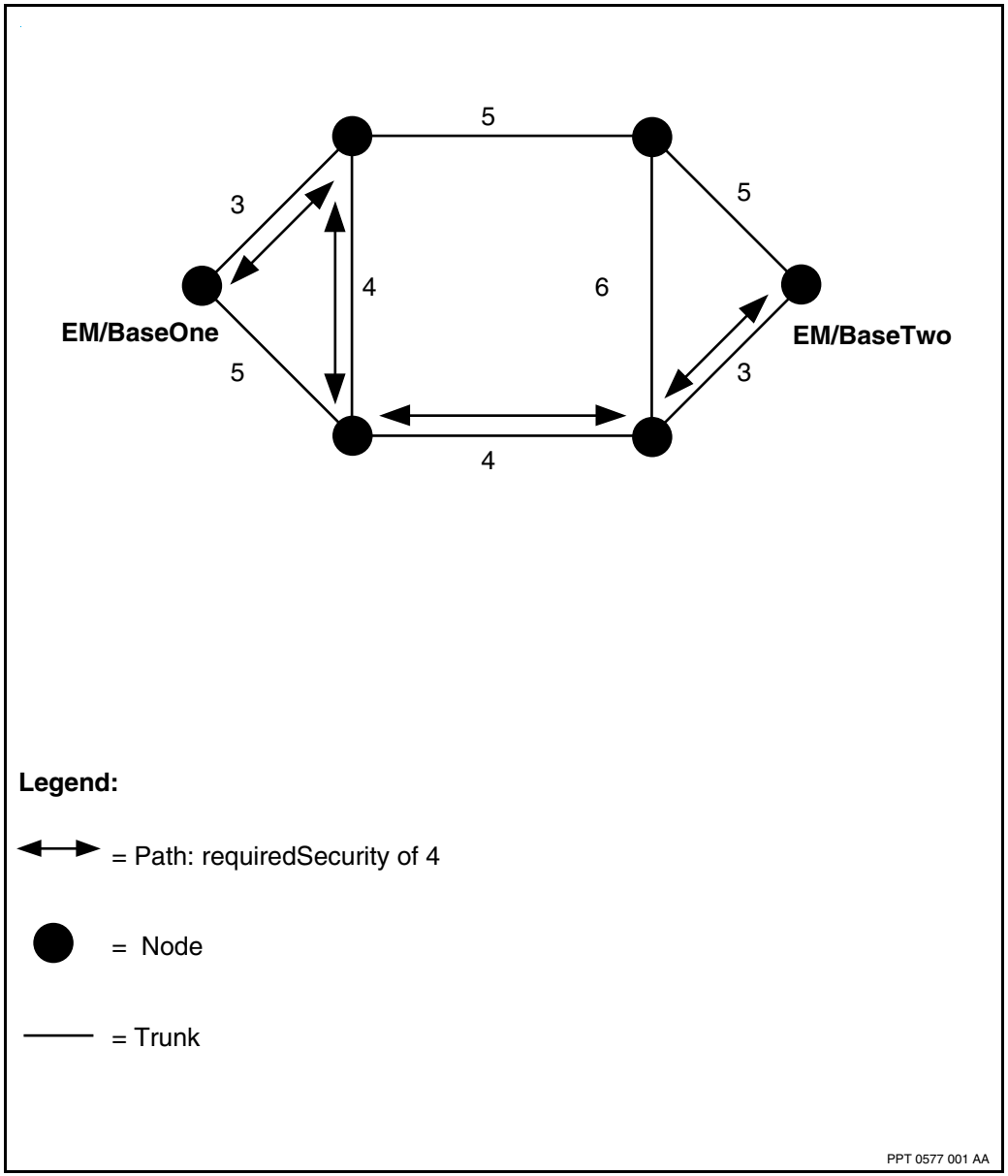
It may be convenient to be able to restrict certain classes of paths to certain Nortel Networks Multiservice Switch trunks. Most of the commonly used qualifiers are represented in security and traffic type. This is an additional option, to be used for any function that you deem appropriate.

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 58).

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.

Figure 15
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 Networks 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.

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

setUpPriority, holdingPriority, and flagsBetweenFrames

These attributes are reserved for future use.

Table 2
Path and Trunk attributes under the HTDS Permanent Logical Connection 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)	
(Sheet 1 of 2)		

Table 2 (continued)
Path and Trunk attributes under the HTDS Permanent Logical Connection component

Path requirement attribute	Trunk attribute	Trunk is candidate for route if...
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.
manualPath (value = list of trunk component IDs)	*trunk component name* (for example: EM/NODER9 trunk/202)	The *Trunk* component name is one of *manualPath*
(Sheet 2 of 2)		

Chapter 3
HDLC Transparent Data Service engineering guidelines

This chapter describes engineering aspects that affect the HDLC Transparent Data Service and illustrates their effect on the operating characteristics of this service. The topics are

Services required by the HDLC Transparent Data Service

The HDLC Transparent Data Service is supported by the Nortel Networks Multiservice Switch Path Oriented Routing System (PORS). PORS provides the underlying routing system for the HDLC Transparent Data Service in a Nortel Networks 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 39) 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

  Note: 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

  Note: There is a maximum combined throughput for the eight interfaces of the V.11 FP. See NN10600-170 *Nortel Networks 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

  Note: 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)

  Note: 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 Networks 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

  Note: The 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 Networks 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

Chapter 4
Component monitoring

This chapter defines the component-monitoring process, which allows reliable operation of the HDLC Transparent Data Service. It covers the following topics:

- how to gather service information
- how to interpret service information

Refer to the following documents for more information on monitoring of the HDLC Transport Data Service.

- NN10600-551 *Nortel Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference*
- NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*
- NN10600-050 *Nortel Networks Multiservice Switch 7400/15000/20000 Command Reference*
- NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*

About components

A component name consists of a sequence of component_type/ component_instance_value pairs. For example, HTDS/5 is a component name.

The component type is the component without a particular instance value. For example, HTDS is a type since it specifies a particular type of component, but does not specify a particular HTDS instance on the node.

The component instance value is the unique name (under a particular component type) which is used to specify the name of particular instance of a component class. For example, in HTDS/5, 5 is the instance value.

Note that the term component class is also used in some of the commands and command examples in this chapter. Component class is a concatenation of component types. For example, HTDS PLC is a component class because it does not contain any instance values.

Viewing components, subcomponents and attributes

Use these common Nortel Networks Multiservice Switch commands to view components, subcomponents and attributes. The commands contained in the procedure use examples applicable to the HTDS service.

1 List all top-level components of the HTDS component which exist in the current view. The "/*" wildcard is permitted for components that are not replicated.

```
list -c HTDS/*
```

2 List all subcomponents of a particular component (for example HTDS/1) while you are in provisioning mode.

```
list HTDS/1
```

3 Display all provisioned attributes of the HTDS component.

```
display HTDS/<htds_number> Framer
```

4 Display a single attribute of a component.

```
display HTDS/<htds_number> Framer crcError
```

5 Display selected attributes of a component by separating the attribute names with commas. For example, display usage state and remote end of all trunks on a module.

```
display trk/* usage, remote
```

6 Display a provisioned attribute of all existing components of a particular component type.

```
display HTDS/* customerIdentifier
```

7 Display all the operational attributes of a component (from the current view).

```
display -o HTDS/<htds_number>
```

8 Display all the provisionable attributes of a component (in the current view).

```
display -p HTDS/<htds_number>
```

9 Lock the HTDS component.

```
lock HTDS/<htds_number>
```

Note: When using this command on a PORS connection, locking the port does not shut down the connection and free the bandwidth. Only locking the service will shut down the connection and free the bandwidth.

Variable definitions

Variable	Definition
<htds_number>	The instance number of the Htds component.

HTDS operational attributes

This section lists the HTDS operational attributes and contains examples which allow the user to display and provision HTDS information.

Table 3
HTDS operational attributes

OSI state	HTDS framer	HTDS Lco
adminState	*frmFromIf*	*state*
usageState	*frmToIf*	*path* ***Note:*** Should the network be divided into clusters and/or topology regions, it should be noted that the *path* attribute can only display information about the current cluster or region segment. For example, should a service traverse a cluster or inter-region link, the *path* attribute shall indicate termination at the cluster or region gateway respectively, and not at the service end point.
proceduralStatus	*aborts*	*end*
alarmStatus	*crcErrors*	*costMetric*
unknownStatus	*lrcErrors*	*delayMetric*
operationalState	*nonOctetErrors*	*roundTripDelay*
availabilityStatus	*overruns*	*setupPriority*
controlStatus	*underruns*	*holdingPriority*
standbyStatus	*largeFrmErrors*	*requiredTxBandwidth*
	normPrioLinkUtilToIf	*requiredRxBandwidth*
	normPrioLinkUtilFromIf	*requiredTrafficType*
		permittedTrunkTypes
		requiredSecurity
		requiredCustomerParameter
		emissionPriority

(Sheet 1 of 2)

Table 3 (continued)
HTDS operational attributes

OSI state	HTDS framer	HTDS Lco
		discardPriority *pathType* *retryCount* *pathFailureCount* *lastTearDownReason* *pktsToNetwork* *bytesToNetwork* *pksFromNetwork* *bytesFromNetwork* *dump*
(Sheet 2 of 2)		

The following examples show some of the operational attributes you can use to monitor the network:

To display information on the *framer* attribute type the command:

```
d HTDS/<n> framer
```

The following information is displayed:

```
Htds/<n> Framer
adminState=unlocked
operationalState=enabled
usageState=busy
frmToIf=0
frmFromIf=0
aborts=0
crcErrors=0
lrcErrors=0
nonOctetErrors=0
overruns=0
underruns=0
```

```
largeFrmErrors=0
normPrioLinkUtilToIf=0%
normPrioLinkUtilFromIf=0%
```

Type this command.

d HTDS/<n> lco

The following information is displayed:

```
Htds/<n> Lco
state = pathUp
path = "EM/BaseOne Htds/4", "EM/BaseSix Htds/12",(etc)
end = called
costMetric = 0
delayMetric = 0 ms
roundTripDelay = 0 ms
setupPriority = 2
holdingPriority = 2
requiredTxBandwidth = 64000 bit/s
requiredRxBandwidth = 64000 bit/s
requiredTrafficType = data
permittedTrunkTypes = terrestrial satellite tt1 tt2
requiredSecurity = 4
requiredCustomerParameter = 4
emissionPriority = 1
discardPriority = 2
pathType = normal
retryCount = 1
pathFailureCount = 0
lastTearDownReason = trunkFailure
pktsToNetwork = 0 packets
bytesToNetwork = 0 bytes
pksFromNetwork = 0 packets
bytesFromNetwork = 0 bytes
```

To display the provisioning of the *Plc* component type the command:

```
d -p HTDS/<n> Plc
```

where:
<n> is the instance value of the *HTDS* component that you used in the view.

The system displays information in the following format:

```
Htds/<n> Plc
remoteName = em/NODEY29E Htds/4
setupPriority = 2
bumpPreference          = bumpWhenNecessary
holdingPriority = 2
requiredTxBandwidth = 64000 bit/s
requiredRxBandwidth = 64000 bit/s
requiredTrafficType = data
permittedTrunkTypes = terrestrial satellite tt1
tt2
requiredSecurity = 4
requiredCustomerParm = 4
pathAttributeToMinimize = cost
maximumAcceptableCost = 1280
maximumAcceptableDelay = 100000 ms
emissionPriority = 1
discardPriority = 2
pathType = normal
manualPath = 0 : ""
1 : ""
2 : ""
3 : ""
4 : ""
5 : ""
6 : ""
7 : ""
8 : ""
9 : ""
pathFailureAction = reRoutePath
optimization = enabled
```

HTDS related Trunk operational attributes

This section lists the operational attributes under the HTDS *Trunk* component and contains examples which allow the user to display and provision *Trunk* component information.

Table 4
Some useful HTDS-related Trunk and Trunk Path Administrator attributes

Trunk	Trunk Path Administrator
adminState	*state*
operationalState	*usedLc*
usageState	*negotiatedMaxLc*
availabilityStatus	*rxPackets*
proceduralStatus	*rxBytes*
controlStatus	*totalCurrentReservedBwOut*
alarmStatus	*currentReservedBwOut*
standbyStatus	*pathSetupCount*
unknownStatus	*pathFailCount*
remoteComponentName	*pathClearCount*
measuredSpeedToIf	*pathBumpCount*
measuredRoundTripDelay	
maxTxUnit	
pktFromIf	
discardUnforward	
intPktFromIf	
discardIntUnforward	

Type the following command:

```
d Trk/<n>
```

The following information is displayed:

```
Trk/<n>
   adminState = unlocked
   operationalState = enabled
   usageState       = busy
   availabilityStatus =
   proceduralStatus   =
   controlStatus      =
   alarmStatus        =
   standbyStatus      = notSet
   unknownStatus      = false
   remoteComponentName = EM/NODER1D TRUNK/0027
   measuredSpeedToIf   = 1918000 bit/s
   measuredRoundTripDelay = 8 ms
   maxTxUnit              = 32768 byte
   pktFromIf              = 1465
   trunkPktFromIf         = 1189
   trunkPktToIf           = 1190
   discardUnforward       = 1
   discardTrunkPktFromIf  = 0
   intPktFromIf           = 437087
   discardIntUnforward    = 0
```

Type the following command:

d Trk/<n> pa

The following information is displayed:

```
Trk/<n> Pa
state = up
usedLc = 1
negotiatedMaxLc = 4096
maxReservableBwOut = 640000 bit/s
unreservedBwOut = 320000 bit/s
reservedBwOutByHp = 2 : 320000 bit/s
attemptCountByHp = 2 : 10
failCountByHp =
clearCountByHp = 2 : 9
bumpCountByHp =
```

Chapter 5
Troubleshooting

This section provides guidelines on what steps you can take to solve problems that may occur after you have installed HTDS. After reading the following sections, you will understand how to react quickly and effectively to these problems. Also included are provisioning checklists, which help ensure a successful HTDS setup. See the following sections for more information:

- "Provisioning checklists" (page 75)
- "Alarms" (page 80)
- "Problem solving" (page 81)

Provisioning checklists

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

- "Installation" (page 75)
- "Provisioning with default values" (page 76)
- "Provisioning optional features" (page 76)

Installation

Make sure you have done the following:

- Do you have the correct software level loaded on every Nortel Networks Multiservice Switch 7400 node that could be a path candidate?
- Have you installed PORS and HTDS according to the instructions in NN10600-270 *Nortel Networks 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 have 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 have done the following:

- See the checklist in the section entitled "Provisioning with default values" (page 76). 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.
- Several of the optional attributes have provisioning dependencies. Check the table below.

Attributes in the same row are interdependent. Provision them accordingly.

Table 5
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.
		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.
(Sheet 1 of 4)		

Table 5 (continued)
Optional attributes

PLC attributes	Trunk attributes	HDLC Framer attributes
	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		
(Sheet 2 of 4)		

Table 5 (continued)
Optional attributes

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.		
(Sheet 3 of 4)		

Table 5 (continued)
Optional attributes

PLC attributes	Trunk attributes	HDLC Framer attributes
emissionPriority Defines which cells or frames are sent first. This has important implications for network engineering.		
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.		
(Sheet 4 of 4)		

Alarms

Alarms are messages used to indicate faults or failure conditions on the node.

Alarms are generated asynchronously by Nortel Networks Multiservice Switch components. When a component generates an alarm, it does so to signal that it is in need of repair or that it has detected a fault elsewhere on the node.

Alarms contain a relatively large amount of information, all of which will assist you in the monitoring and surveillance of your network. Because alarms are such an important and integral part of Multiservice Switch 7400 fault management, they are described separately in document NN10600-500 *Nortel Networks Multiservice Switch 6400/7400/15000/20000 Alarms Reference.*

Causes of alarms

As a general rule, you can expect to see an alarm in the following situations:

- degradation/quality-of-service conditions, (for example if a threshold is reached)
- processing errors (for example protocol violations)
- failures/out-of-service conditions (for example, hardware or facility failures)
- administrative conditions (for example, the lock command is issued)
- security violations

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

Problem solving

Problems with setting up connections may be due to errors or mismatches in setting up the system or provisioning. Use the “Provisioning checklists” (page 75) 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?

The following flowchart may help you to determine the answers to these questions.

Figure 16
Flowchart: an example of troubleshooting using the LCo component

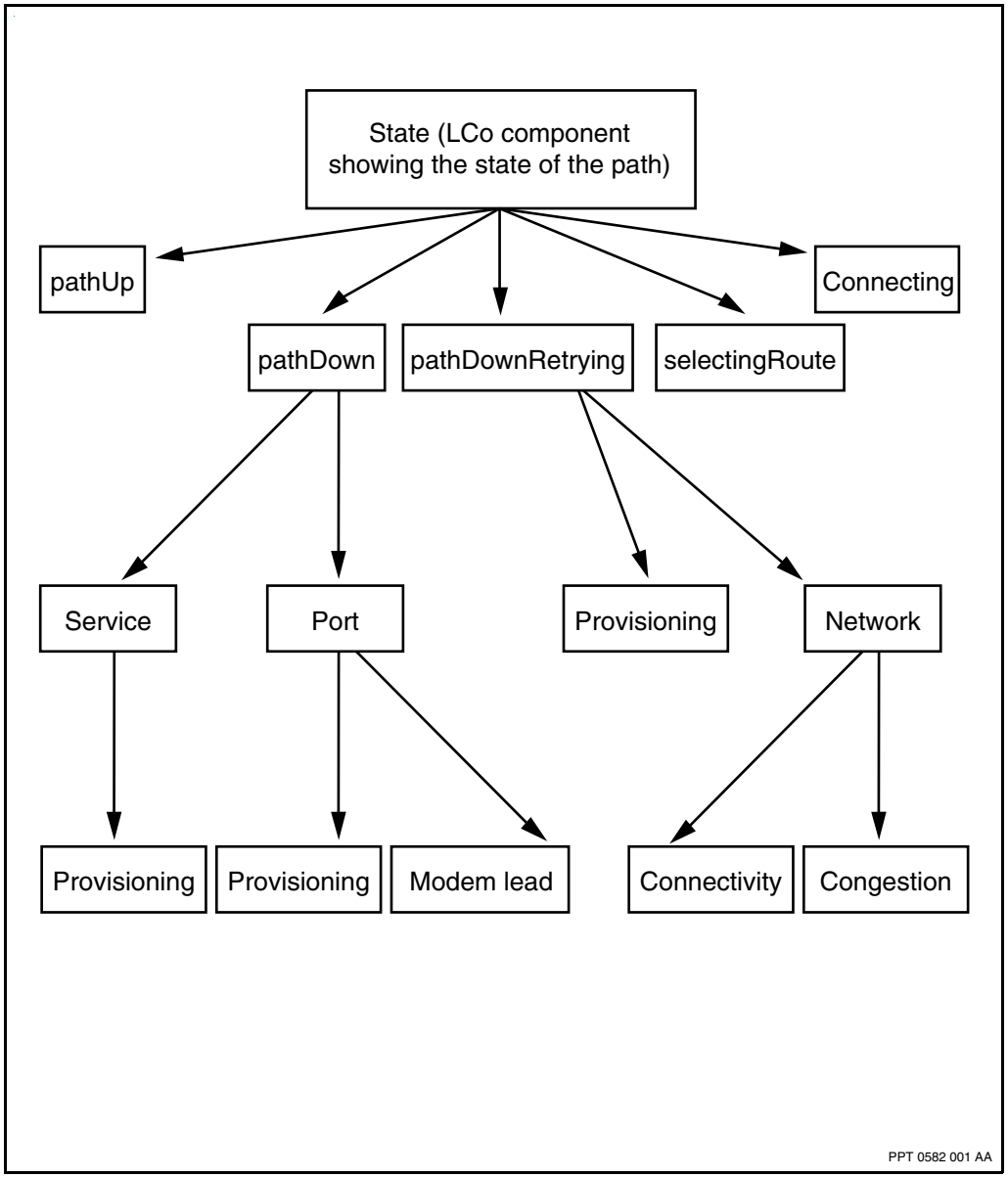


Table 6
Handling problems

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 Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference* and NN10600-550 *Nortel Networks Multiservice Switch 7400/ 15000/20000 Common Configuration Procedures*.
(Sheet 1 of 4)		

Table 6 (continued)
Handling problems

Problems that may occur	Probable causes	Corrective measures
The connection doesn't come up.	Physical connections are not secure.	Check the cable, connections, and pins. See NN10600-175 *Nortel Networks Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*. For a description of an FP see NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*. To install an FP see NN10600-175 *Nortel Networks Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*. To provision an FP see NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* and NN10600-551 *Nortel Networks 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 75). Trunk provisioning information must be identical for both ends of the connection. Reprovision with correct information.
(Sheet 2 of 4)		

Table 6 (continued)
Handling problems

Problems that may occur	Probable causes	Corrective measures
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 Networks Multiservice Switch 7400/ 15000/20000 FP Configuration Reference* and NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for provisioning information.
Note: If a problem has occurred before with this connection, check the *lastTeardownReason* attribute under the *Plc* component.		
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.
(Sheet 3 of 4)		

Table 6 (continued)
Handling problems

Problems that may occur	Probable causes	Corrective measures
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 Networks 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 29).

(Sheet 4 of 4)

Checking service configuration

Check the configuration of the HTDS service.

1. To obtain information about the settings of the provisioned view of the *Plc* component, from the provisioning mode, type:

   `d -p htds/<htds_number> Plc`

2. Print this information, if possible, as you will be comparing it to other provisioning data.

3. To obtain information about the settings of the current view of the Trunk Path Administrator component, from the provisioning mode, type:

   `d -p trunk/<trunk_number> Pa`

4. Use the tables “Optional attributes” (page 77) and “Path and Trunk attributes under the HTDS Permanent Logical Connection component” (page 59) to help you determine how the attributes should relate.

Variable definitions

Variable	Definition
<htds_number>	The instance number of the Htds component in this view.
<trunk_number>	The instance number of the trunk in this view.

Checking bandwidth

Obtain information about the bandwidth used and the bandwidth available.

1 Display bandwidth information from Trunk Path Administrator:

```
d -p trk/<trunk_number> pa
```

2 Display bandwidth information from Routing Route Selector:

```
d rtg rs
```

3 Determine if the amount of bandwidth reserved is greater than the amount available.

Variable definitions

Variable	Definition
<trunk_number>	The instance number of the trunk in this view.

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 Route Selector 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. See "HTDS related Trunk operational attributes" (page 73) for more information on the results of displaying a component.

Example 2

```
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.

Nortel Networks Multiservice Switch 7400

Operations: HDLC Transparent Data Service

Release 6.1

Publication: NN10600-770
Document status: Standard
Document version: 6.1S1
Document date: August 2004
Printed in Canada

NORTEL NETWORKS