NORTEL NETWORKS

Nortel Networks Multiservice Switch 7400

Operations: Bit Transparent Data Service

NN10600-775

Nortel Networks Multiservice Switch 7400

Operations: Bit Transparent Data Service

Publication: NN10600-775
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 NN10600-775 *Nortel Networks Multiservice Switch 7400 Operations: Bit Transparent Data Service* describes the Passport Bit Transparent Data Service (BTDS).

The following topics are discussed in this section:

Who should read this document

The NN10600-775 *Nortel Networks Multiservice Switch 7400 Operations: Bit Transparent Data Service* is for persons with the following BTDS responsibilities:

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

- troubleshooting

What you need to know

The NN10600-775 *Nortel Networks Multiservice Switch 7400 Operations: Bit Transparent Data Service* assumes that you have an understanding of Nortel Networks Multiservice Switch 7400 network architecture.

The terms service or BTDS used in this document refer to the Multiservice Switch 7400 Bit Transparent Data Service.

The term MVP-E refers to multipurpose voice platform enhanced echo cancellation (MVP-E) function processors.

How this document is organized

The NN10600-775 *Nortel Networks Multiservice Switch 7400 Operations: Bit Transparent Data Service* contains the following sections:

- "Bit Transparent Data Service configuration" (page 19) provides detailed procedures for provisioning the Bit Transparent Data Service in your network.
- "BTDS application examples" (page 35) contains examples which describe how to provision the Bit Transparent Data Service for typical applications.
- "Bit Transparent Data Service (BTDS) fundamentals" (page 49) is an overview of the Bit Transparent Data Service (BTDS) and Path Oriented Routing System (PORS).
- "Establishing connections—route selection" (page 73) is a continuation of the discussion of PORS, looking at the optional features.
- "BTDS implementation guidelines" (page 85) outlines the aspects of Bit Transparent Data Service engineering that must be addressed to ensure that the service operates efficiently.
- "Component monitoring" (page 93) describes common operational mode commands, and how to use them for monitoring BTDS components and attributes.
- "Troubleshooting" (page 101) outlines how to troubleshoot the BTDS 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*.
- "Structural changes" (page 15)
- Updated this document to remove references to these MVP FPs which are support discontinued (SDed):
  - 1-port DS1 MVP with cardtype 1pDS1MVP and PEC NTFN62
  - 1-port DS1Voice with cardtype 1pDS1V and PEC NTFP41
  - 1-port E1 MVP with cardtype 1pE1MVP and PEC NTFN20
  - 1-port E1Voice with cardtype 1pE1V and PEC NTFP43
  - 1-port J2MV with cardtype J2MV and PEC NTBP96
  - 1-port TTC2M MVP with cardtype 1pTTC2mMVP and PEC NTFN64

Structural changes

This document was restructured into a modular, task-based format to improve the usability of the information. The following changes were made to this document:

- Procedures were grouped into higher-level tasks.
- Task flow charts were added to improve navigation through tasks and procedures, to set tasks and procedures in context, and to provide a visual representation of prerequisites and configuration paths.
- Procedures were restructured into a modular format.
- Purpose statements were added to tasks and procedures to provide context.

- Prerequisites were divided into those applicable to an entire task. those applicable only to a specific procedure, and those applicable only to a specific procedure step. Prerequisites applicable to an entire task were placed in the appropriate task-level prerequisite section, prerequisites applicable only to a specific procedure were placed in the prerequisite section of the procedure, and prerequisites applicable only to a specific step were placed in the step.
- 'Where' statements were removed from procedures and the content placed in the 'Variable values' table following the procedure.
- A 'Procedure Job Aid' section was added to procedures where appropriate. This consists of information that supports the procedure, such as a component hierarchy figure, a checklist, or a diagram.
- Conceptual and reference information were removed from procedures, placed in the appropriate conceptual or reference section, and cross-referenced from the procedure where appropriate.

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

  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 Passport documentation library, see NN10600-001 *Nortel Networks Multiservice Switch 7400/15000/20000 Basics: Customer Documentation*. See also the *Nortel Networks Multiservice Switch Release Notes Supplement*, which provides information on last minute software updates for particular releases.

In particular, refer to the following documents for information on installing and operating BTDS in your network:

- NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*

- NN10600-175 *Nortel Networks Multiservice Switch 7400 Hardware Installation, Maintenance, and Upgrade*
- NN10600-270 *Nortel Networks Multiservice Switch 7400/15000/20000 Software Installation*
- NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*
- NN10600-551 *Nortel Networks Multiservice Switch 7400/15000/20000 FP Configuration Reference*
- NN10600-050 *Nortel Networks Multiservice Switch 7400/15000/20000 Command Reference*
- NN10600-750 *Nortel Networks Multiservice Switch 7400 Operations: Voice Transport*
- NN10600-435 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*
- NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*
- NN10600-500 *Nortel Networks Multiservice Switch 6400/7400/15000/ 20000 Alarms Reference*

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
Bit Transparent Data Service configuration

Provision BTDS using the default settings provided with the package, or you can provision it to meet your specific requirements. The default settings are designed to provide a connection made up of the least number of hops across the network.

Prerequisites to Bit Transparent Data Service configuration

- Use the procedures in NN10600-270 *Nortel Networks Multiservice Switch 7400/15000/20000 Software Installation* to install BTDS software.
- Ensure that the version of your routing software contains the Path Oriented Routing System (PORS). BTDS 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.
- When installing BTDS, you must set the *featureList* attribute appropriately:

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

- Ensure that the logical processor type for trunks has PORS software in its feature list. Use the following command:

  `set software lpt/trunk featureList porsTrunks`

Note: All nodes in the network that are candidates for BTDS traffic must be running the latest software and must have the Trunk pathAdmin component added to the view.

- Provision network clock synchronization. Before beginning the provisioning process, map out how you want the master clocking signal to be distributed through the network. See NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for information on how to structure the delivery of the clocking signal and how to configure network clock synchronization.
- See NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference* for complete descriptions of BTDS components and attributes and available values.
- Refer to NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for basic provisioning procedures. For information on commands, refer to NN10600-050 *Nortel Networks Multiservice Switch 7400/15000/20000 Command Reference*.

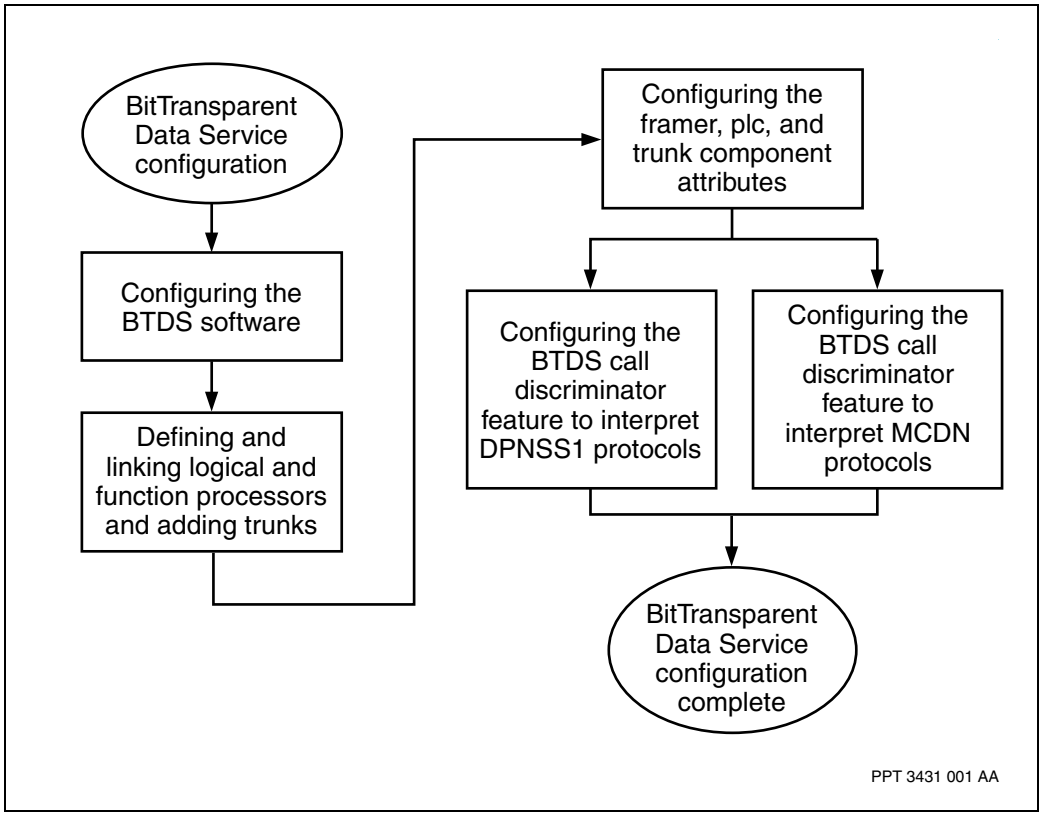
If you require hardware installation information when installing BTDS, refer to the following documents:

- 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*

Bit Transparent Data Service configuration task

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

Figure 1
Bit Transparent Data Service configuration task flow



Task navigation

- "Configuring the BTDS software" (page 23)
- "Defining and linking logical processors and adding trunks" (page 24)
- "Configuring the attributes of the Framer, Plc, and Trunk components" (page 27)
- "Configuring the BTDS call discriminator feature to interpret DPNSS1 protocols" (page 31)
- "Configuring the BTDS call discriminator feature to interpret MCDN protocols" (page 33)

Configuring the BTDS software

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

Procedure steps

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

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

2. Set the *featureList* attribute to vtds to add the BTDS software.

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

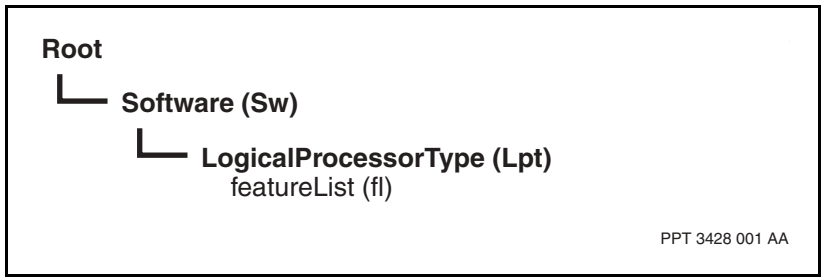
3. Repeat step 1 and step 2 to configure BTDS software on the remote end node.

Variable definitions

Variable	Definition
<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 BTDS software component hierarchy



Defining and linking logical processors and adding trunks

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

Procedure steps

1. Add logical processors.

   add lp/<lp_number>

2. Link the LPs to the configured BTDS software.

   set lp/<lp_number> lpt software lpt/<name>

3. Propagate the new settings throughout the module.

   activate prov

4. Define the Bit Transparent Data Service instance.

   add btds/<btds_number>

5. Define the *interfaceName* attribute to link the Framer component to the hardware.

   set btds/<btds_number> framer interfaceName lp/ <lp_number> <port_type>/<port_number>

6. Define the *Plc* component's *remoteName* attribute.

   set btds/<btds_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. Set up the interrupting trunk feature.

   set trunk/<trunk_number> unack framer framingType interrupting

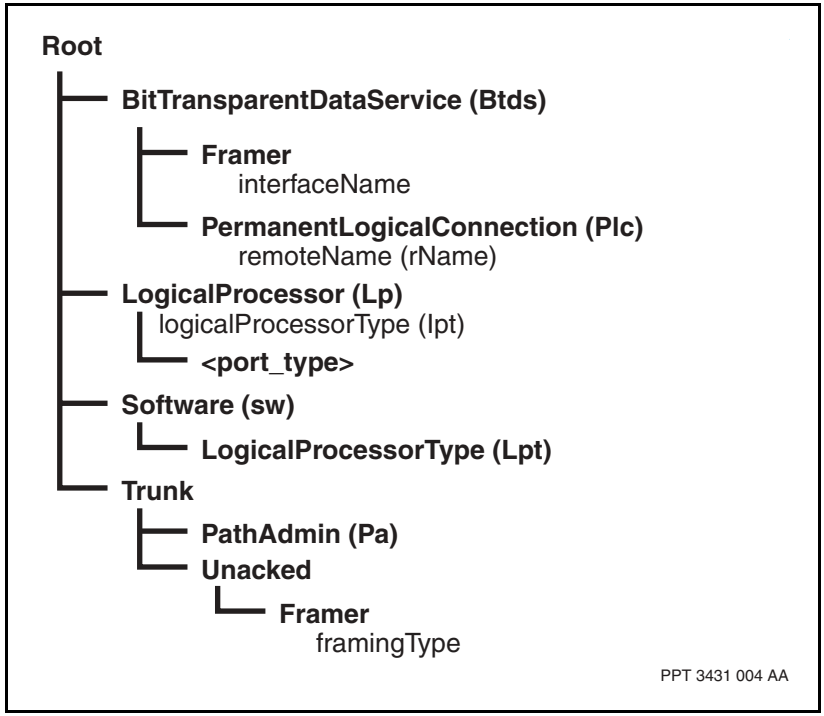
10. Repeat step 1 through step 9 to define and link logical processors and to add trunks to the other end of the connection.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds 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 BTDS software (for example, TDS).
<port_number>	The port number 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 and service name exactly (not case-sensitive), or a connection will not be made, even if the source and destination FPs share compatible provisioning data. If you do not specify a value for the *remoteName* attribute, the *Btds Plc* component accepts connection requests from any remote *Btds Plc* component. However, with no value specified for the *remoteName* attribute, the *Btds Plc* component cannot originate a connection request and the connection establishment process is slower.
<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 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 btds/<btds_number> framer <framer_attribute> <framer_attribute_value>

2. Define values for the attributes that you want to change for the *Plc* component.

   set btds/<btds_number> plc <plc_attribute> <plc_attribute_value>

3. Define values for the *Trunk* component attributes.

   set trunk/<trunk_number> unack framer <unacked_framer_attribute> <unacked_framer_attribute_value>

4. Define values for *Trunk PathAdmin* component's attributes.

   set trunk/<trunk_number> pathAdmin <trunk_attribute> <trunk_attribute_value>

5. Repeat step 1 through step 4 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. To use the SPO-mux or SPO-map mode of transport, set the *btdsCellSize* attribute to 44 bytes. To use the AAL5 mode, set the *btdsCellSize* attribute to 84 bytes. DS1, E1, V.11, and V.35 FPs support the larger BTDS cell size of 84 bytes. If you are using fewer than four timeslots (256 Kbps) on a DS1 or E1 FP, set the *btdsCellSize* attribute to the default value of 44 bytes. If you are using more than three timeslots, set the *btdsCellSize* attribute to the larger BTDS cell size of 84 bytes. If you are setting up a BTDS connection between a voice and MVP-E FP, refer to "Voice and MVP-E function processor interworking restrictions" (page 91) before defining values for the preceding attributes.
<btds_number>	The instance number of the Btds 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. Set the *pathType* attribute to manual or forced.
<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.
<unacked_framer_attribute>	One of the provisionable attributes of the Unacked Framer component.
<unacked_framer_attribute_value>	The value of the Unacked Framer component's provisionable attribute.

Procedure job aid

Figure 4
Configuring the Framer and Plc components attributes component hierarchy

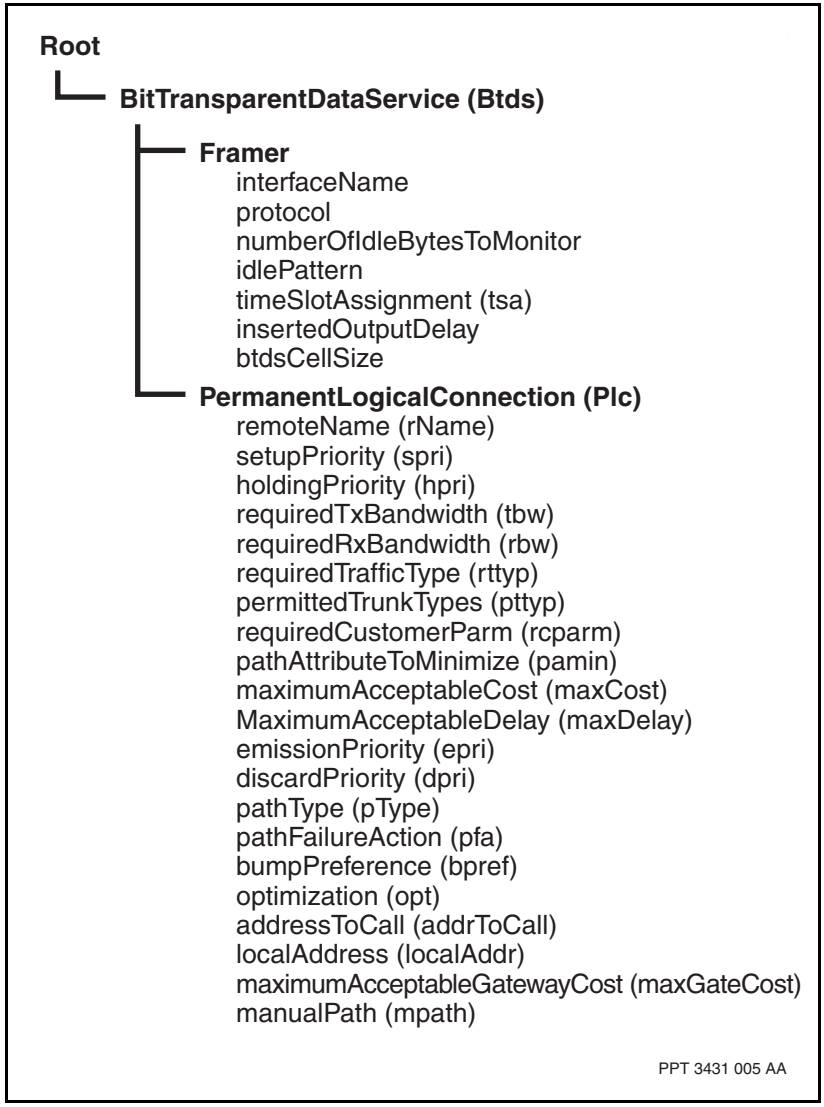
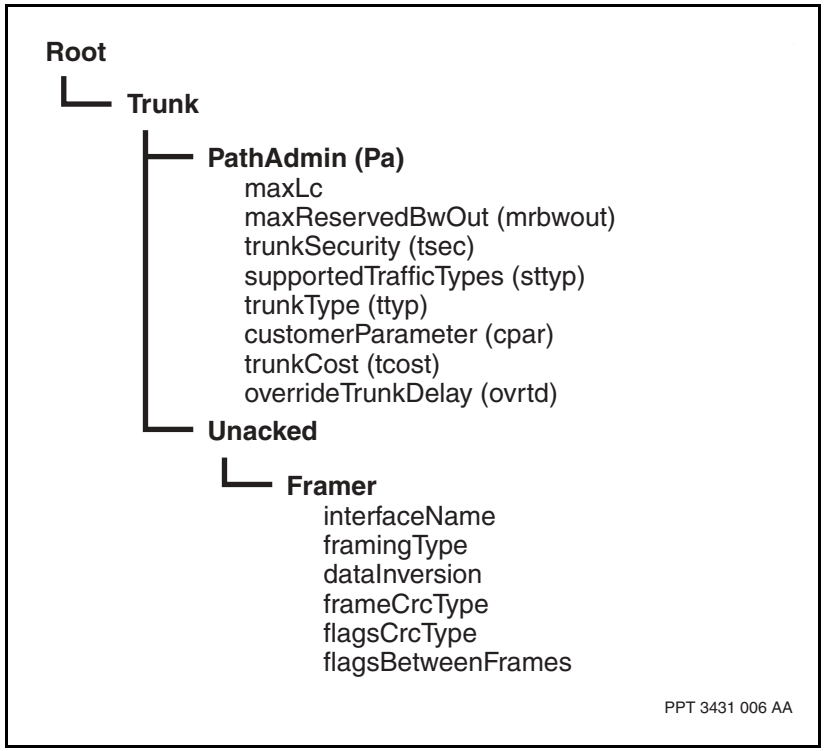


Figure 5
Configuring the Trunk component attributes component hierarchy



Configuring the BTDS call discriminator feature to interpret DPNSS1 protocols

You can provision the call discriminator feature on Voice Transport connections between voice FPs that use a CCS protocol. The supported CCS protocols are DPNSS1 and MCDN (see "Configuring the BTDS call discriminator feature to interpret MCDN protocols" (page 33)).

Procedure steps

1. Set the *featureList* attribute to callDiscriminator.

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

2. Define the CCS handler component to specify that DPNSS 1 protocol messages will be handled for the purpose of voice/data call discrimination.

   ```
   add btds/<btds_number> Dpnss1
   ```

3. Define the Dpnss1 component's *timeslotsX* attribute. Provisioning of the *timeslotsX* attribute must match the provisioning of the timeslots at the PBX which is connected to the voice FP. For example, if timeslots 1, 2, 3, 4 are provisioned X in the PBX, then type the following command:

   ```
   set btds/<btds_number> Dpnss1 timeslotsX 1 2 3 4
   ```

   This attribute specifies that timeslots 1, 2, 3 and 4 are allocated a priority X or master priority for collision resolution.

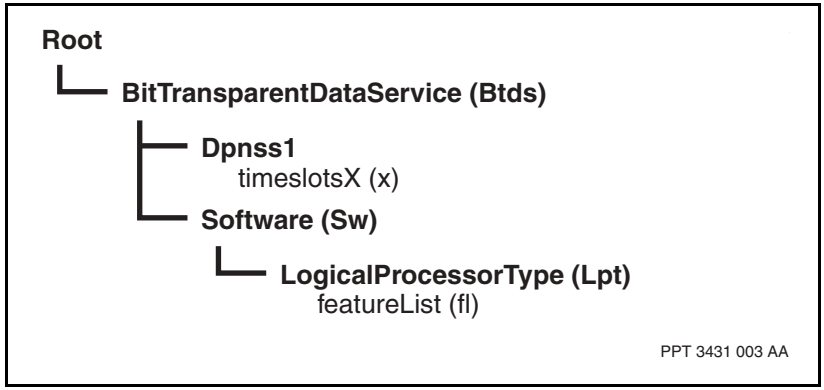
   Note: For consistency of call collision resolution, the timeslotsX for the voice FP located at the far end must be different than the *timeslotsX* attribute provisioned at the near end.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<name>	The name you assigned to the BTDS software (for example, TDS).

Procedure job aid

Figure 6
Configuring the BTDS call discriminator feature to interpret DPNSS1 protocols component hierarchy



Configuring the BTDS call discriminator feature to interpret MCDN protocols

You can provision the call discriminator feature on Voice Transport connections between voice FPs that use a CCS protocol. The supported CCS protocols are DPNSS1 (see "Configuring the BTDS call discriminator feature to interpret DPNSS1 protocols" (page 31)) and MCDN.

Procedure steps

1 Set the *featureList* attribute to callDiscriminator.

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

2 Define the CCS handler component.

```
add btds/<btds_number> Mcdn
```

3 Define the Mcdn component's *adjPbxSide* attribute.

```
set btds/<btds_number> Mcdn adjPbxSide master
```

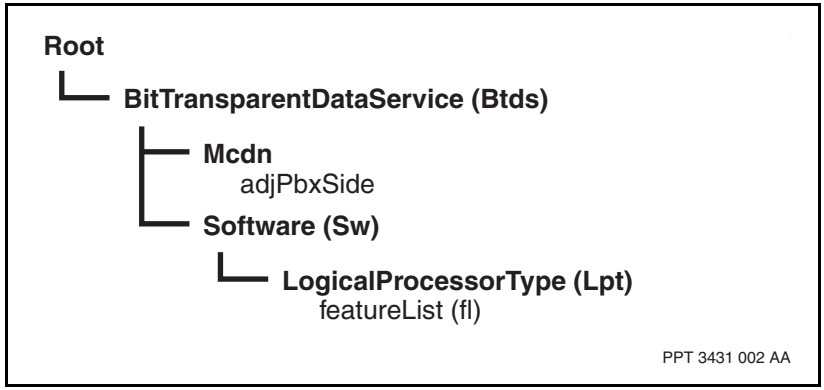
This attribute specifies master priority for call collision resolution. Its value must match the adjacent PBX, the one at the other end of the primary rate interface (PRI). The default setting of this attribute is slave. The corresponding *Mcdn* component, at the other end of the btds connection, should have its *adjPbxSide* attribute set to slave.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<name>	The name you assigned to the BTDS software (for example, TDS).

Procedure job aid

Figure 7
Configuring the BTDS call discriminator feature to interpret MCDN protocols component hierarchy



Chapter 2
BTDS application examples

The following sections provide examples of how to configure BTDS, and include how to configure idle suppression:

- "Configuring a Common Channel Signalling (CCS) link on a DS1/E1 MVP-E FP" (page 36)
- "Configuring idle suppression to suppress NRZI-encoded HDLC frames on E1 MVP-E and DS1 MVP-E FPs" (page 39)
- "Configuring video applications" (page 41)
- "Configuring video-on-demand servers" (page 43)

See also "Setting the overrideRemoteName attribute" (page 45).

"Provisioning notes" (page 47) contains details about setting certain attributes in the following procedures. Also, see "Configuring the remoteName attribute" (page 91) for important end-to-end negotiation information before proceeding to the following application examples.

Configuring a Common Channel Signalling (CCS) link on a DS1/E1 MVP-E FP

Use this procedure when configuring a CCS link on a DS1/E1 MVP-E FP.

Note: TTC2M MVP-E FPs do not support CCS protocols.

Prerequisites

- Refer to "Voice and MVP-E function processor interworking restrictions" (page 91) if you plan to establish a CCS link between a voice and MVP-E FP.
- Review the information in the table "Provisioning notes" (page 47).
- The *timeslotAlignment* attribute must be set to unaligned on MVP-E FPs because they do not support frame aligned BTDS. See "Voice and MVP-E function processor interworking restrictions" (page 91) for more details.

Procedure steps

1. Add the FP to the shelf.

   `set shelf card/<card_number> cardType <card>`

2. Provision a logical processor to run on the FP.

   `add lp/<lp_number>`

3. Link the LP to the FP.

   `set lp/<lp_number> mainCard shelf card/<card_number>`

4. Define the software to be loaded.

   `set lp/<lp_number> logicalProcessorType sw lpt/<name>`

5. Add a port to the FP.

   `add lp/<lp_number> <port_type>/<port_number>`

6. Set the linetype.

   `set lp/<lp_number> <port_type>/<port_number> lineType <framing_format>`

7 Set the zeroCoding attribute.

```
set lp/<lp_number> <port_type>/<port_number>
zeroCoding <zeroCoding_value>
```

8 Set the raiAlarmType attribute.

```
set lp/<lp_number> <port_type>/<port_number>
raiAlarmType <raiAlarmType_value>
```

9 Include the card in network clock synchronization.

```
set lp/<lp_number> <port_type>/<port_number>
clockingSource module
```

10 Set a timeslot. There is only one timeslot per channel for a Bit Transparent Data Service on a CCS link.

```
set lp/<lp_number> <port_type>/<port_number> chan/
<signaling_channel> timeslots <timeslot_number>
```

11 Define the Bit Transparent Data Service instance.

```
add btds/<btds_number>
```

12 Set the remote name of the BTDS application at the far end of the Nortel Networks Multiservice Switch 7440 network.

```
set btds/<btds_number> plc remoteName <remote_name>
```

13 Set the required bandwidth.

```
set btds/<btds_number> plc requiredRxBandwidth
<requiredRxBandwidth_value>
```

14 Link this BTDS application with the desired timeslot.

```
set btds/<btds_number> framer interfaceName lp/
<lp_number> <port_type>/<port_number> chan/
<signaling_channel>
```

15 To configure idle suppression, set the *protocol* attribute to hdlc. CCS links are all HDLC-based.

```
set btds/<btds_number> framer protocol hdlc
```

16 Repeat step 1 through step 15 to configure the CCS link at the other end of the connection.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<card>	One of 1pDs1Mvpe or 1pE1Mvpe.
<card_number>	The instance number of the card.
<framing_format>	The type of framing pattern for the line. The signaling protocol you specify determines what type of framing format to use.
<lp_number>	The instance number of the LP.
<name>	The name you assigned to the BTDS software.
<port_number>	The port number you assigned when you provisioned the port.
<port_type>	Either ds1 (for 1pDs1Mvpe) or e1(for 1pE1Mvpe).
<raiAlarmType_value>	The value for the method of sending a remote alarm indication (RAI) alarm on the line.
<remote_name>	The PLC's remote end address; the name of the other end of the connection.
<requiredRxBandwidth_value>	As a general rule, you should set the required bandwidth of the PLC to three times the actual CCS frame rate.
<signaling_channel>	The channel you assign to carry signaling information.
<timeslot_number>	The timeslot number.
<zeroCoding_value>	The value for zero encoding on the line.

Configuring idle suppression to suppress NRZI-encoded HDLC frames on E1 MVP-E and DS1 MVP-E FPs

Use this procedure to suppress NRZI-encoded HDLC frames on E1 MVP-E and DS1 MVP-E FPs.

NRZI encoding means that a transition only occurs when a zero is encountered. That is, if the pattern 0111 1110 (7E hex) is NRZI encoded, then it would appear on the line as either 0000 0001 or 1111 1110, depending on whether the value on the previous line is a 1 or a 0.

Prerequisites

- Review the information in the table "Provisioning notes" (page 47).

Procedure steps

1 Follow the steps below. For more information on each of the steps, refer to the similar scenario described in procedure "Configuring a Common Channel Signalling (CCS) link on a DS1/E1 MVP-E FP" (page 36).

```
set shelf card/<card_number> cardType <card>

add lp/<lp_number>

set lp/<lp_number> mainCard shelf card/<card_number>

set lp/<lp_number> logicalProcessorType sw lpt/<name>

add lp/<lp_number> <port_type>/<port_number>

set lp/<lp_number> <port_type>/<port_number> lineType
<framing_format>

set lp/<lp_number> <port_type>/<port_number>
clockingSource module

set lp/<lp_number> <port_type>/<port_number> chan/
<signaling_channel> timeslots <timeslot_number>

add btds/<btds_number>

set btds/<btds_number> plc remoteName <remote_name>

set btds/<btds_number> plc requiredTxBandwidth
<requiredTxBandwidth_value>
```

```
set btds/<btds_number> plc requiredRxBandwidth
<requiredRxBandwidth_value>
```

2 Provision the framer to suppress NRZI-encoded HDLC data.

```
set btds/<btds_number> framer interfaceName lp/
<lp_number> <port_type>/<port_number> chan/
<signaling_channel>
```

```
set btds/<btds_number> framer protocol nrziHdlc
```

3 Repeat step 1 through step 2 to configure the other end of the connection.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<card>	The type of FP.
<card_number>	The instance number of the card.
<framing_format>	The type of framing pattern for the line. The signaling protocol you specify determines what type of framing format to use.
<lp_number>	The instance number of the LP.
<name>	The name you assigned to the BTDS software.
<port_number>	The port number you assigned when you provisioned the port.
<port_type>	The type of port.
<remote_name>	The PLC's remote end address; the name of the other end of the connection.
<requiredRxBandwidth_value>	The required value for the requiredRxBandwidth attribute is based on the assumption that the actual data rate will be less than one-third of the provisioned value (that is, 10,000 bits/second).
<requiredTxBandwidth_value>	The required value for the requiredTxBandwidth attribute is based on the assumption that the actual data rate will be less than one-third of the provisioned value (that is, 10,000 bits/second).
<signaling_channel>	The channel you assign to carry signaling information.
<timeslot_number>	The timeslot number.

Configuring video applications

Use this method for any video application. It is the only method that works for applications where the idle pattern can appear in the data stream (that is, transparent bisynchronous).

Prerequisites

- Review the information in the table "Provisioning notes" (page 47).
- The *timeslotAlignment* attribute must be set to unaligned on MVP-E FPs because they do not support frame aligned BTDS. See "Voice and MVP-E function processor interworking restrictions" (page 91) for more details.

Procedure steps

1 Follow the steps below. For more information on each of the steps, refer to the similar scenario described in procedure "Configuring a Common Channel Signalling (CCS) link on a DS1/E1 MVP-E FP" (page 36).

```
add lp/<lp_number>

add lp/<lp_number> <port_type>/<port_number>

set lp/<lp_number> <port_type>/<port_number> linkMode
dce

set lp/<lp_number> <port_type>/<port_number>
clockingSource module

add btds/<btds_number>

set btds/<btds_number> plc remoteName <remote_name>

set btds/<btds_number> plc requiredTxBandwidth
<requiredTxBandwidth_value>

set btds/<btds_number> plc requiredRxBandwidth
<requiredRxBandwidth_value>
```

2 Configure the BTDS for universal idle frame suppression. The BTDS must detect 1000 consecutive frames containing the identical 16-bit data pattern before it suppresses idle frames at the ingress node.

```
set btds/<btds_number> framer interfaceName lp/
<lp_number> <port_type>/<port_number>
```

```
set btds/<btds_number> framer protocol universal

set btds/<btds_number> framer numOfIdleBytesToMonitor
1000
```

3 Repeat step 1 through step 2 to configure the other end of the connection.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<lp_number>	The instance number of the LP.
<port_number>	The port number you assigned when you provisioned the port.
<port_type>	The type of port.
<requiredRxBandwidth_value>	The required value for the requiredRxBandwidth attribute is based on the assumption that the actual data rate will be less than one-third of the provisioned value (that is, 10,000 bits/second).
<requiredTxBandwidth_value>	The required value for the requiredTxBandwidth attribute is based on the assumption that the actual data rate will be less than one-third of the provisioned value (that is, 10,000 bits/second).

Configuring video-on-demand servers

Use this procedure to configure video-on-demand servers.

Prerequisites

- Clock synchronization is very important for the Bit Transparent Data Service. Set *linkMode* to dte only if the network surrounding Nortel Networks Multiservice Switch 7440 is synchronized.
- Review the information in the table "Provisioning notes" (page 47).
- MVP-E FPs do not support the *timeslotAlignment* attribute setting of unaligned. See "Voice and MVP-E function processor interworking restrictions" (page 91) for more details.

Procedure steps

1 Follow the steps below. For more information on each of the steps, refer to the similar scenario described in procedure "Configuring a Common Channel Signalling (CCS) link on a DS1/E1 MVP-E FP" (page 36).

```
start prov

add lp/<lp_number>

add lp/<lp_number> <port_type>/<port_number>

set lp/<lp_number> chan/<signaling_channel> timeslots
<timeslot_number>

set lp/<lp_number> <port_type>/<port_number> lineType
<framing_format>

set lp/<lp_number> <port_type>/<port_number>
zeroCoding <zeroCoding_value>

set lp/<lp_number> <port_type>/<port_number>
clockingSource module

add btds/<btds_number>

set btds/<btds_number> plc remoteName <remote_name>

set btds/<btds_number> framer interfaceName lp/
<lp_number> <port_type>/<port_number>

set btds/<btds_number> framer timeslotAlignment
frameAligned
```

2 Repeat step 1 to configure the other end of the connection.

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<framing_format>	D4 or ESF Use ESF with B8ZS to avoid potential transmission problems if the video signal transmits a long stream of zeroes. For an E1 voice function processor, the lineType attribute should be set to CCS. The type of framing pattern for the line. The signaling protocol you specify determines what type of framing format to use.
<lp_number>	The instance number of the LP.
<port_number>	The port number you assigned when you provisioned the port.
<port_type>	The type of port.
<remote_name>	The PLC's remote end address; the name of the other end of the connection.
<signaling_channel>	The channel you assign to carry signaling information.
<timeslot_number>	The value used for the timeslots can be any positive number. For example, 1, 2, 3, and so on. The actual number of timeslots provisioned is dependent on the desired bandwidth.
<zeroCoding_value>	AMI or B8ZS

Setting the overrideRemoteName attribute

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

Prerequisites

- Review the information in the table "Provisioning notes" (page 47).

Procedure steps

1 Follow the steps below.

```
add btds/<btds_number>

add lp/<lp_number> <port_type>/<port_number>

set lp/<lp_number> <port_type>/<port_number> chan/
<signaling_channel> timeSlots <timeslot_numbers>

set lp/<lp_number> <port_type>/<port_number>
clockingSource module

set lp/<lp_number> <port_type>/<port_number> chan/
<signaling_channel> applicationFramerName btds/
<btds_number> framer

set btds/<btds_number> framer interface lp/<lp_number>
<port_type>/<port_number> chan/<signaling_channel>

set btds/<btds_number> plc remoteName <remote_name>
```

2 Repeat step 1 to configure the other end of the connection.

3 Set the *overrideRemoteName* attribute.

```
set btds/<btds_number> lc overrideRemoteName
<override_name>
```

Variable definitions

Variable	Definition
<btds_number>	The instance number of the Btds component.
<lp_number>	The instance number of the LP.

(Sheet 1 of 2)

Variable	Definition
<override_name>	An override name for the PLC's remote end-point's address. The setting of the overrideRemoteName overrides the setting for the remoteName attribute.
<port_number>	The port number you assigned when you provisioned the port.
<port_type>	The type of port.
<remote_name>	The PLC's remote end address; the name of the other end of the connection.
<signaling_channel>	The channel you assign to carry signaling information.
<timeslot_numbers>	The timeslot numbers.

(Sheet 2 of 2)

Provisioning notes

The information in the table "Provisioning notes" (page 47) gives you details about setting some of the *Framer* component attributes.

Table 1
Provisioning notes

Framer component attribute	Details
protocol	You can set the *protocol* attribute to hdlc on voice and MVP-E FPs to configure BTDS to suppress ingress HDLC idle frames. A frame is an HDLC idle frame if each byte is an identical, non-inverted HDLC idle byte. The setting of the *protocol* attribute to hdlc specifies that BTDS suppress ingress idle frames when two or more consecutive frames match the preceding description (up to 88 bytes).
protocol	You can set the *protocol* attribute to nrziHdlc on voice and MVP-E FPs to configure BTDS to suppress ingress non-return to zero inverted (NRZI) HDLC idle frames. A frame is an NRZI-encoded HDLC idle frame if each byte is an identical, NRZI-encoded HDLC idle byte. The setting of the *protocol* attribute to nrziHdlc specifies that BTDS suppress ingress idle frames when two or more consecutive frames match the preceding description (up to 88 bytes).
protocol	You can set the *protocol* attribute to universal on voice and MVP-E FPs to configure BTDS to suppress all ingress idle frames. A frame is a universal idle frame if each 16-bit word is identical. The setting of the *protocol* attribute to universal specifies that BTDS suppress ingress idle frames after a specific number of consecutive frames matches this description (specified under the *numberOfIdleBytesToMonitor* attribute).
numberOfIdleBytesToMonitor	The *numberOfIdleBytesToMonitor* attribute can be set on voice and MVP-E FPs. This attribute is only used when the *protocol* attribute is set to universal. The value of the *numberOfIdleBytesToMonitor* attribute specifies the number of consecutive, identical ingress 16-bit words that must occur before suppressing ingress idle data.Set the value of this attribute to a number higher than the largest frame that can be sent into this service.
(Sheet 1 of 2)	

Table 1 (continued)
Provisioning notes

Framer component attribute	Details
idlePattern	The value contained under the *idlePattern* attribute specifies the hexadecimal byte pattern to use to fill an egress frame locally when the BTDS suppresses ingress idle frames at a remote node. The type of idle suppression you configure determines the value to specify under the *idlePattern* attribute. The default value is FF Hex.
insertedOutputDelay	The provisionable range for the *insertedOutputdelay* attribute is from 0 to 50 milliseconds with a default value of 12 milliseconds. To aid the user in choosing the required setting for the *insertedOutputdelay* attribute, the *frmUnderRuns* and *frmLostInNetwork* attributes can be used. If there is an increase in the *frmUnderRuns* and no corresponding increase in *frmLostInNetwork*, then this indicates that the cross network delay is varying. Increasing the value for the *insertedOutputDelay* attribute will compensate for this change. If there is an increase in the *frmLostInNetwork* value with increasing *frmUnderRuns* value, then a potential network problem may exist. It is possible for underruns to occur if the inserted delay time is not large enough to compensate for cross network delay variations.
timeslotAlignment	The Framer attribute *timeslotAlignment* enables or disables timeslot alignment on a BTDS connection. You cannot set *timeslotAlignment* to frameAligned on an MVP-E FP. If frame alignment is not required, then the *timeslotAlignment* attribute should be set to unaligned, because frame alignment does impact performance (due to variable packet lengths used to achieve alignment).
(Sheet 2 of 2)	

Chapter 3
Bit Transparent Data Service (BTDS) fundamentals

The following sections provide an overview of the Nortel Networks Multiservice Switch 7440 Bit Transparent Data Service (BTDS):

- "What is the Bit Transparent Data Service?" (page 49)
- "How does the Bit Transparent Data Service work?" (page 50)
- "Establishing connections using PORS" (page 59)
- "Tips for configuring BTDS" (page 69)

For fundamentals on network clock synchronization, see NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*.

What is the Bit Transparent Data Service?

The Bit Transparent Data Service (BTDS) provides a means of transmitting any continuous bit-rate data, such as video or Bisync. The data is routed unchanged (transparently) over a synchronous, end-to-end connection. The service is provided through the use of common Nortel Networks Multiservice Switch trunks and without the need for additional customer equipment.

The BTDS requires network clock synchronization and the Path Oriented Routing System (PORS). Network clock synchronization provides network-wide clock synchronization to ensure accurate transmission and reproduction of data throughout the network. PORS provides automatically selected end-

to-end paths that allow predictable delay variations, and maintain cell ordering within a selected path. A cell, in this document, refers to a small fixed-size data packet.

How does the Bit Transparent Data Service work?

During a call, the BTDS transfers data between two user devices over a permanent logical connection (PLC). The data stream at the source is formatted into fixed length cells and sent across the network. At the destination, the cells are reformatted into the original data stream.

One or more Nortel Networks Multiservice Switch trunks are normally provisioned to allow paths forming a connection based on traffic type, security, bandwidth, delay requirements, cost requirements and other attributes. Since a path is pre-provisioned, all cells pertaining to a connection follow the same route to their common destination. Refer to figure "PLCs, LCs and LCNs" (page 64) for permanent logical connection, logical channel, and logical channel number definitions.

See the following sections for more information about the BTDS:

- "BTDS characteristics" (page 50)
- "Transparent cell transport" (page 52)
- "Suppression of idle data" (page 52)
- "Egress delay control" (page 52)
- "BTDS efficiency" (page 53)
- "Frame alignment and non-frame alignment" (page 57)
- "Call discriminator" (page 58)
- "Resource management" (page 59)

BTDS characteristics

The key characteristics of the BTDS include

- constant bit rate
- bidirectional network-wide synchronous transfer of bit stream by means of fixed-length cells

- transparent cell transport
- suppression of idle frames
- resource management
- provisionable attributes which support the following functionality:
  - BTDS cell size
  - idle pattern
  - protocol support
  - setup, holding, emission, and discard priorities
  - transmit and receive bandwidth
  - route minimization criteria: cost or delay
  - maximum cost or maximum delay values
  - action on path failure
  - egress delay control
  - frame alignment and non-frame alignment
  - call discriminator
- interfaces supported:
  - V.11, V.35, DS1, E1, 1pDS1V, 1pE1V, DS1 MVP-E, E1 MVP-E, TTC2M MVP-E, and 8pDS1

  Note: For 8pDS1 only PM2 is supported, for up to 4 ports.

- interconnections supported include
  - DS1 to DS1 or E1 to E1
  - V.11 to V.11 or V.35 to V.35
  - V.11 or V.35 to DS1 or E1
  - V.11 to V.35
  - voice function processor (FP) to MVP-E FP (See “Voice and MVP-E function processor interworking restrictions” (page 91) for more information.)

— DS1 or 8pDS1 to DS1 or 8pDS1

Transparent cell transport

Data is formed into cells at the source node. Each cell contains the original data and all path information required to reach the destination. All cells are tagged as high priority (by default). The payload portion of each cell is transferred transparently across the network (that is, without being interpreted).

Suppression of idle data

BTDS allows you to suppress idle frames at the ingress side of the Nortel Networks Multiservice Switch 7440 network to help conserve bandwidth in the subnet. BTDS can be provisioned to suppress HDLC-encoded, NRZI-encoded HDLC, or a specific number of identical, repeated idle frames. The *protocol* attribute specifies the ingress frames to be suppressed by the BTDS. For more information about configuring BTDS idle suppression, see "Provisioning notes" (page 47).

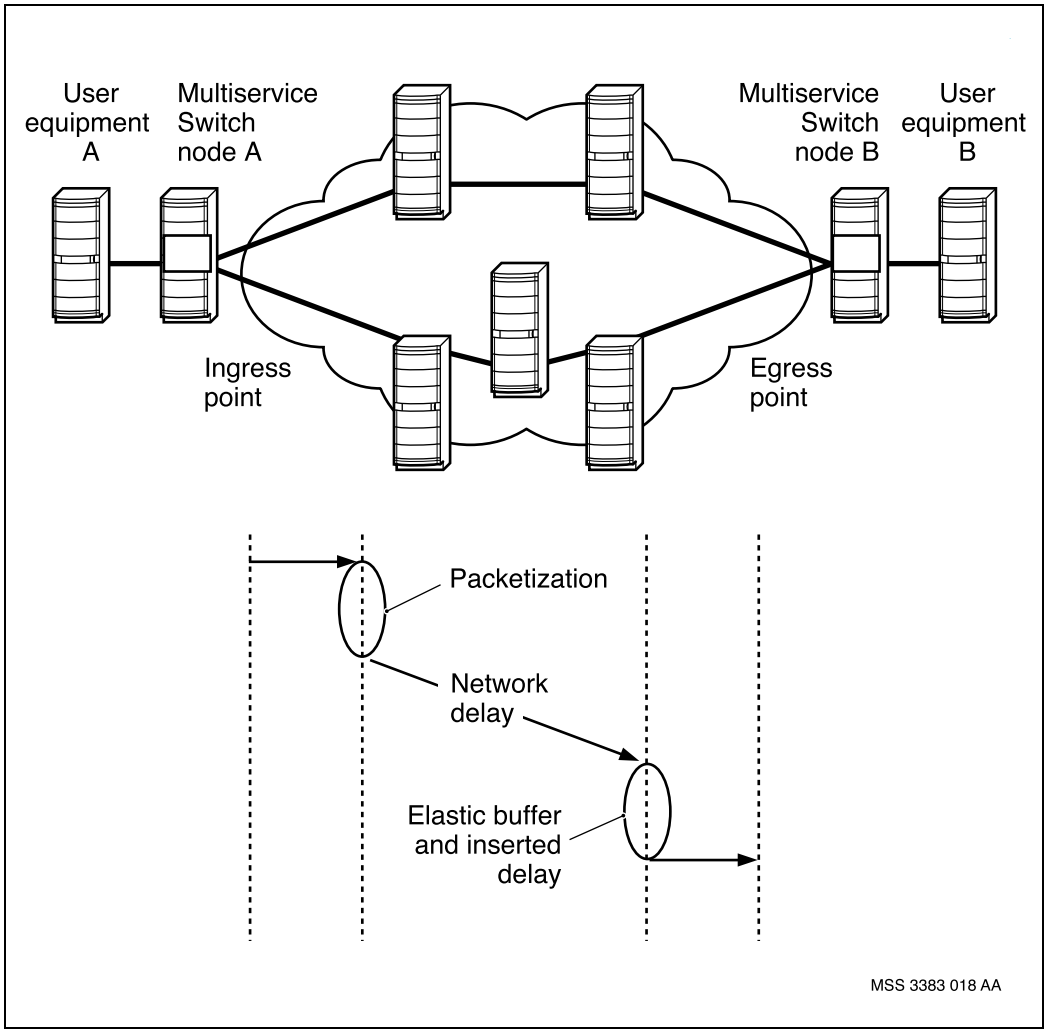
Egress delay control

The egress control capability allows the user to insert a time delay at the egress point of the Nortel Networks Multiservice Switch network. This time delay is used to compensate for variations in cross network delay.

Note: The *btdsCellSize* attribute affects egress time delay. Larger BTDS cell sizes cause longer delays.

Figure "Typical timing delays in a Multiservice Switch network" (page 53) depicts the timing delays associated with a sample network. Timing delay can occur at three separate points, the network interface, through the network and at the network egress point. Timing delay at the network interface is a function of the packetization delay. This delay is dependent on the cell size and the line speed. Delay through the network is dependent on the network topology. Delay at the egress point is a function of the elastic buffer and the inserted delay. The elastic buffer is dependent on the installed hardware and is typically 5 cells in length. The setting for the inserted delay is dependent on the provisionable *insertedOutputDelay* attribute. Setting of this attribute allows the user to adjust the inserted delay for specific applications which are delay sensitive. For more information on setting the *insertedOutputDelay* attribute refer to "Provisioning notes" (page 47).

Figure 8
Typical timing delays in a Multiservice Switch network



BTDS efficiency

The BTDS cell size can be provisioned to optimize bandwidth on Nortel Networks Multiservice Switch trunks over ATM (formerly known as ATM logical trunks). You can provision a larger cell size to take advantage of the

PORS ATM Adaptation Layer 5 (AAL5) transport mode, or the smaller cell size for the PORS short path-oriented multiplexing (SPO-mux) mode or SPO-map mode. See the following sections for more information on cell size:

- “Large cell size” (page 54)
- “Small cell size” (page 56)

Large cell size

The larger BTDS cell size provides the following benefits:

- increased BTDS throughput on DS1, E1, V.11, and V.35 FPs
- efficient transportation across frame-cell and Multiservice Switch trunks over ATM

Figure “AAL5 PDU carrying 44-byte BTDS cells” (page 55) shows 44 byte BTDS cells on AAL5 trunks. The 84-byte BTDS cell size on AAL5 trunks is shown in figure “AAL5 PDU carrying 84-byte BTDS cells” (page 56). With an increased BTDS cell size of 84 bytes, BTDS uses the full ATM cell.

Larger BTDS cells cause larger minimum egress delays—as the size of the BTDS cell increases, the time to accumulate the BTDS cell increases. This increase in time directly impacts the minimum egress delay that BTDS can achieve. At higher line speeds, the impact of a larger BTDS cell size on minimum egress queue delay decreases.

When setting up the BTDS efficiency feature, please note the following restrictions:

- increased BTDS cell size is valid only on DS1, E1, V.11, and V.35 FPs
- below 4 timeslots, BTDS cell sizes larger than 44 bytes are not valid. In this case, use the smaller cell size and SPO-mux transport mode.

Figure 9
AAL5 PDU carrying 44-byte BTDS cells

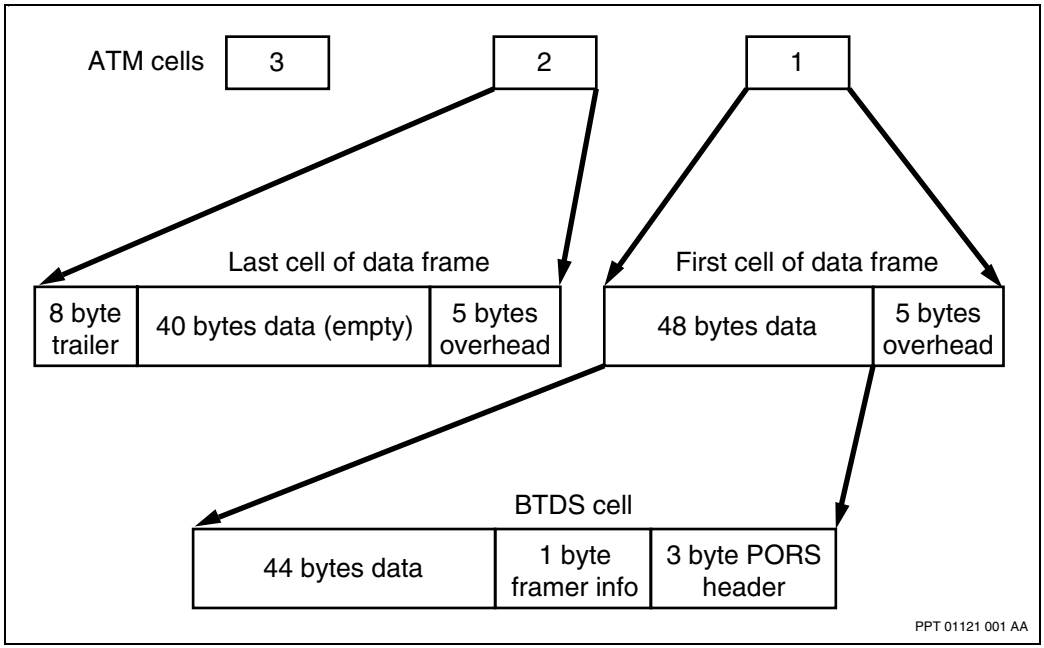
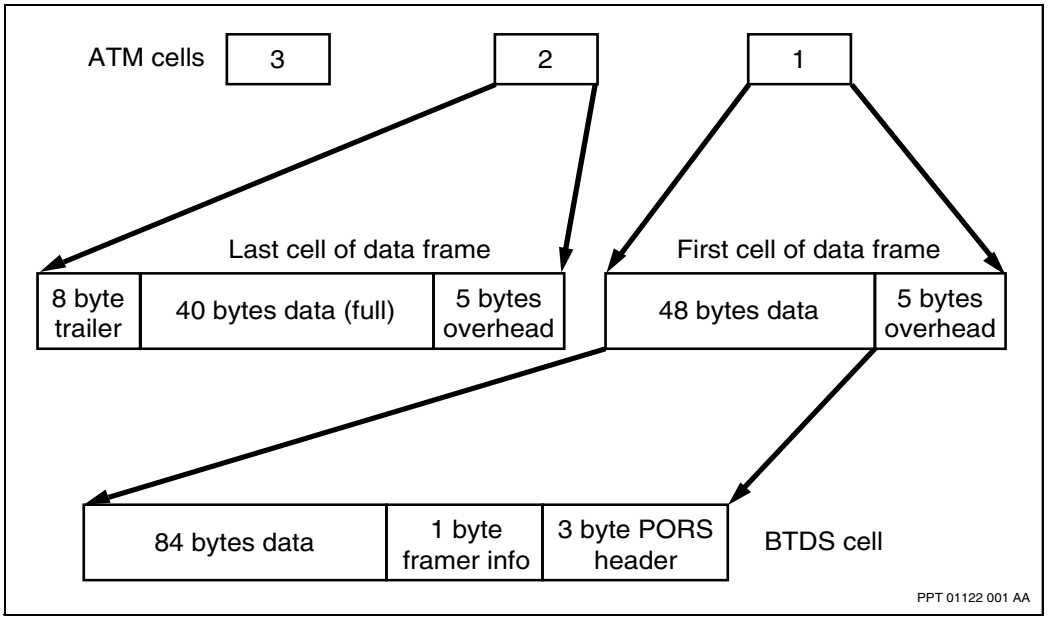


Figure 10
AAL5 PDU carrying 84-byte BTDS cells



Small cell size

With cells of 44 bytes, you can provision the smaller cell size and use the SPO-mux or SPO-map transport mode. (If the connection is using less than four DS1 or E1 timeslots, you must use the smaller cell size.) In SPO-mux or map mode, the 44-byte BTDS cells are packaged with the 3-byte PORS header and the 5-byte overhead, to use 53 bytes. These frames are then transported as pure cells, without the need for AAL5 adaptation.

If the Multiservice Switch trunk over ATM selected through PORS does not support SPO-mux or map mode, the cells are transported using AAL5 adaptation.

For more information on cell transport modes, see NN10600-420 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Trunking*.

Frame alignment and non-frame alignment

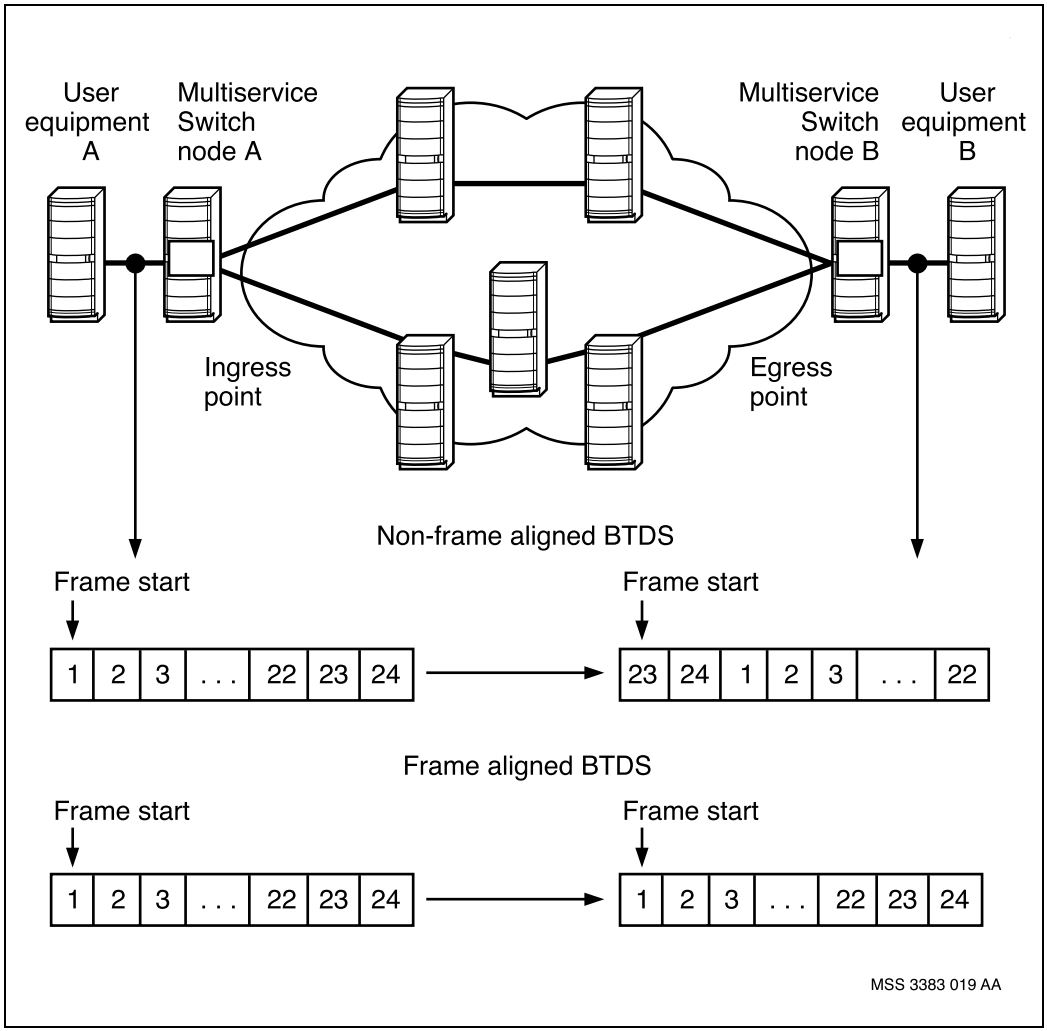
BTDS supports both frame aligned and non-frame aligned traffic. Frame or timeslot alignment is defined as the preservation of the timeslot positioning within a data stream. Frame alignment is used with customer equipment which is sensitive to the timeslot in which the data is transmitted. Figure "Frame aligned or non-frame aligned BTDS in a Multiservice Switch network" (page 58) shows the differences between the two types of traffic.

For non-frame aligned BTDS, data from User Equipment A is sent into the Nortel Networks Multiservice Switch subnet in the following order: 1, 2, 3, through to 22, 23, and 24. Since the timeslot of the transmitted data is not preserved, the data could be sent out of the subnet to User Equipment B in the following order: 23, 24, 1, 2, 3 through to 20, 21, 22. For frame-aligned traffic, the timeslot is preserved. For the scenario shown in figure "Frame aligned or non-frame aligned BTDS in a Multiservice Switch network" (page 58), the timeslots within the data stream sent into the subnet by User Equipment A will be preserved. This ensures that the timeslots within the data stream received by User Equipment B will match the timeslots sent by User Equipment A. For the case discussed previously, the order would be 1, 2, 3, through to 22, 23, and 24.

Frame alignment and non-frame alignment is provisioned by the *timeslotAlignment* attribute. Setting of this attribute to *unaligned* ensures the timeslots will not be preserved through the network. Setting the attribute to *frameAligned* ensures that the timeslots will be sent transparently through the Multiservice Switch network unchanged. Details on provisioning the frame-aligned BTDS is contained in "BTDS application examples" (page 35).

> ***Note:*** Only voice FPs support frame alignment. If frame alignment is enabled on a voice FP, then no other services can be provisioned on that FP, and it cannot interwork with an MVP-E FP.

Figure 11
Frame aligned or non-frame aligned BTDS in a Multiservice Switch network



Call discriminator

Only voice function processors (FPs) support the call discriminator capability. The call discriminator capability allows a Nortel Networks Multiservice Switch node to differentiate from either a voice or data call received from a PBX and to automatically alter the operational mode of the

voice service depending on the type of call. The node is able to dynamically determine the call type (either data or voice) by handling CCS messages based on Digital Private Network Signalling System No 1 protocol (DPNSS 1) and Meridian 1 ISDN Primary Rate Interface protocol (MCDN). This capability is also described in NN10600-750 *Nortel Networks Multiservice Switch 7400 Operations: Voice Transport*. Provisioning of this capability is handled by the BTDS subcomponents, *Dpnss1* or *Mcdn*. For more information on provisioning this capability, refer to "Configuring the BTDS call discriminator feature to interpret DPNSS1 protocols" (page 31) and "Configuring the BTDS call discriminator feature to interpret MCDN protocols" (page 33).

Resource management

To provide BTDS, each end-point BTDS application provides the following major 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 existence of the service
  - 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

Establishing connections using PORS

The Path Oriented Routing System (PORS) sets up a permanent connection between two ends of a network path. The connection is called permanent since it is established with a fixed path (determined by the resources available at call setup) and, barring network difficulties, remains in place until removed. Once the path has been provisioned, the BTDS user may regard it as an end-point to end-point wire.

When the path is provisioned on a node, you will need to provide an instance value under the *Btds* component for each end of the connection (both nodes are provisioned). To complete the connection, the node and instance of the opposite end of the connection are identified in each provisioning session.

Path establishment is automatic. For more information about components see NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*.

See the following sections for more information about PORS connections:

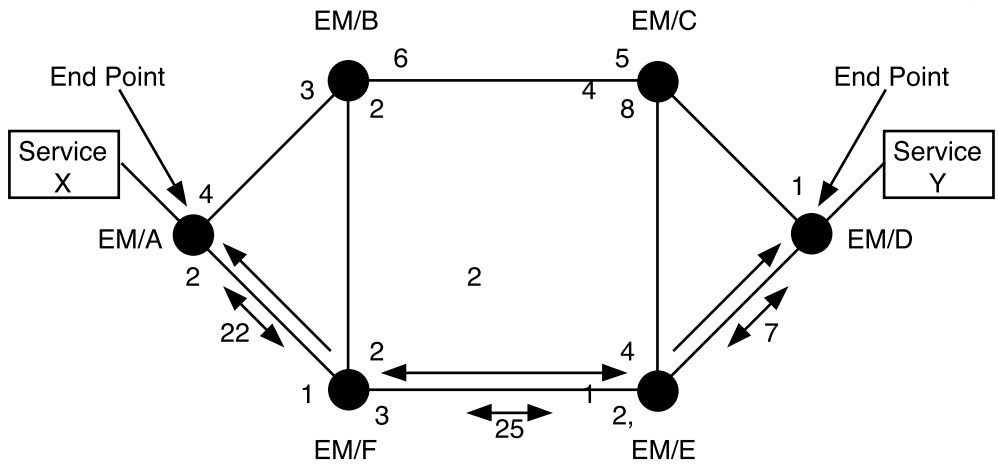
- "Establishing a path" (page 60)
- "Using the default PORS attribute values" (page 62)
- "Creating the path" (page 62)
- "Path bumping" (page 63)
- "Optimizing paths" (page 63)
- "Recovering from path establishment failure" (page 64)
- "Multiservice Switch trunk bandwidth allocation" (page 65)

Establishing a path

Figure "Path and path description" (page 61) shows a path across a six-node network and illustrates some of the terms used in this section.

> ***Note:*** Provisioning data should either be identical at both ends of the connection (for example, some Framer attribute values) or must point to the exact identifier of the other end of the connection (PLC component's *remoteName* attribute). If the other end's name is not correct, no path will be established. This also applies to end-to-end negotiation between voice and MVP-E function processors. See "Voice and MVP-E function processor interworking restrictions" (page 91) for more information.

Figure 12
Path and path description



Legend:

2 = Trunk instance

25 = LCh - Logical Channel Number

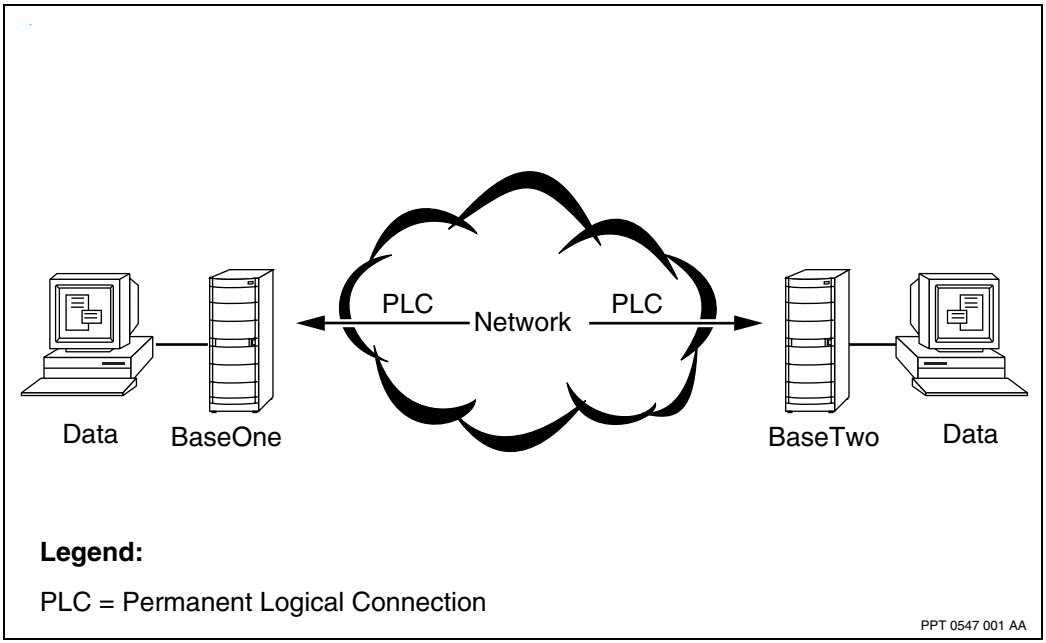
= Node

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 0546 001 AA

Figure 13
Multiservice Switch 7440 path across the network



Using the default PORS attribute values

BTDS comes with a set of default values for most of the attributes associated with the service. The default values are designed to set up a Permanent Logical Connection (PLC) using the optimal route across the network.

In many networks the default values are sufficient; however, you can choose how PORS selects a route. Attributes that influence selection are explained in “Establishing connections—route selection” (page 73).

Creating the path

A 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 all of the Nortel Networks Multiservice Switch trunks that will be necessary for the path. At each point along the route, the following actions are triggered:

- creation of the Logical Channel (LC) 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. 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

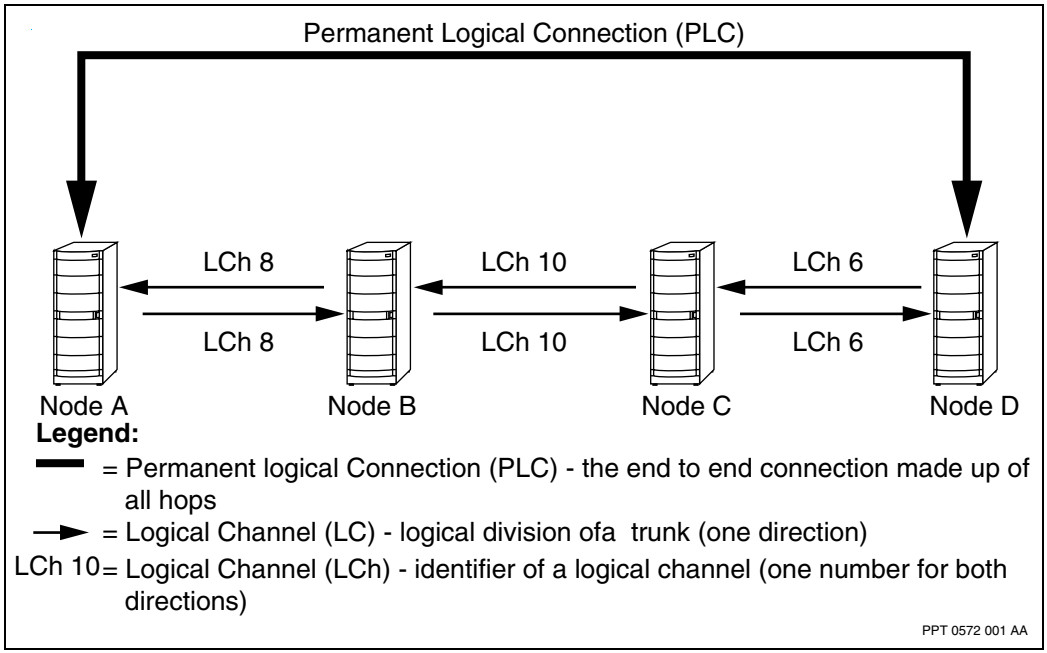
Over time, a PORS connection may end up on a less than optimal path due to link failures, node software upgrades, Nortel Networks Multiservice Switch trunks being locked, or other possible scenarios. Path optimization periodically attempts to move the PORS connection back to a more optimal path. The 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 241-6401-435 *Passport 6400 Path-Oriented Routing System Guide*.

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 command "add rtg pors" to activate this feature on a node.

Figure 14
PLCs, LCs and LCNs



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, FP, or Nortel Networks Multiservice Switch trunk) along the chosen route.
- A Multiservice Switch trunk has reached the maximum number of paths (logical channels) that it can support (*maxLc*).

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

The following sections list provisionable parameters provided by a PLC or Trunk Path Administrator that allow different policies of bandwidth allocation to be enforced:

Actual ranges of values or specific parameters that may be used with the attributes in the following sections are listed in NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference*.

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 Multiservice Switch 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.

When the BTDS Framer attribute *protocol* is set to any value other than *none*, the *requiredRxBandwidth* and *requiredTxBandwidth* attribute should be set to 3 or 4 times the actual data rate. For example, if the actual line rate is 64,000 bit/s, but the offered data rate is 2400 bit/s, the *requiredRxBandwidth* and the *requiredTxBandwidth* attribute should be set to 9600 bit/s.

MaxLc

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

MaxReservedBwOut

This is the percentage of total 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.

The Path Oriented Routing System allows bandwidth reservation to be less than the peak bandwidth requirements. However, to take advantage of the statistical nature of traffic patterns in networks, it is recommended that BTDS bandwidth requirements be matched to the bandwidth consumption since BTDS is a constant bit rate service.

As a general rule, specify 115% of the load offered by the application. For example, if video equipment connected to BTDS uses a 384 kbit/s interface, specify 440 kbit/s as the required bandwidth.

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 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:

Table 2
Example setup and holding priorities

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

Note: The values listed in table "Example setup and holding priorities" (page 67)provide example settings only and are not the default settings.

The *setupPriority* and *holdingPriority* attributes of 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 new path's *setupPriority* is greater
(that is, higher priority) than the existing path's *holdingPriority*.

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 cell will be emitted to the Multiservice Switch trunk. The more urgent the emission priority the sooner the cell is sent to the trunk. Emission and discard priorities are set independently. Some transmissions, voice for example, need a high emission priority but do not need a high discard priority since discarded cells have a low impact on service quality.

The *emissionPriority* and *discardPriority* attributes of the PLC component affect all cells on a particular path. Emission priority reflects the urgency that a cell reach its destination as quickly as possible, while discard priority reflects the importance put on a cell reaching its destination.

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

> ***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.

Using the interrupt queue

Highest emission-priority BTDS cells can be sent through the interrupt queue quickly. Short, uniform BTDS cells from the interrupt queue are placed in the data stream ahead of longer, less urgent frames. The *framingType* attribute of the *Framer sub*component, under *Trunk Unack component*, can be set to allow the use of the interrupt queue. When the *framingType* attribute is set to *interrupt*, the interrupt queue is activated. When the *framingType* attribute is set to *hdlc*, the interrupt queue is deactivated.

> ***Note:*** Trunk components should be set to the same value on both ends of the connection. Failure to do so can result in a failure to obtain a path.

BTDS transmissions can use the Multiservice Switch trunk without the interrupt queue but the quality of service will be reduced.

Specifying that a path terminate and not reroute

Some applications using BTDS may not tolerate the delays caused by rerouting. To cause a path to terminate instead of rerouting, set the *pathFailureAction* attribute of the PLC component to the required value. The default setting is reroutepath.

Tips for configuring BTDS

The following sections contain a set of general guidelines for using BTDS with other types of traffic:

- "Emission and discard priority" (page 69)
- "Other bandwidth considerations" (page 71)

Emission and discard priority

BTDS uses a strict priority system—higher emission-priority cells get the necessary amount of bandwidth faster. For this reason, too much high priority traffic will restrict the flow of low priority traffic. As a general rule, the amount of frame-cell trunks interrupting mode (highest priority) traffic should not be greater than 80% (limit *maxReservedBwOut* attribute to 80% or less).

In general, BTDS traffic should be setup as described in figure "Emission and discard priority setup for BTDS traffic" (page 70). When using video, you should setup BTDS and other traffic as shown in figure "Emission and discard priority setup for BTDS video traffic" (page 71). Make sure to account for loads that are not immediately apparent, such as burstiness and control traffic, when determining the amount of BTDS traffic.

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

Figure 15
Emission and discard priority setup for BTDS traffic

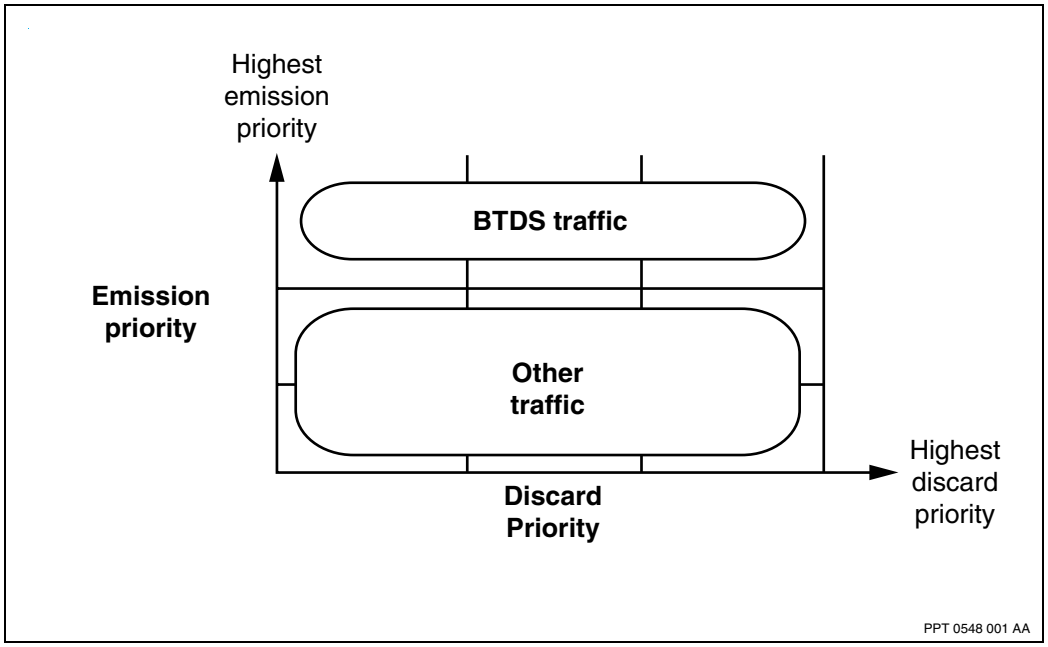
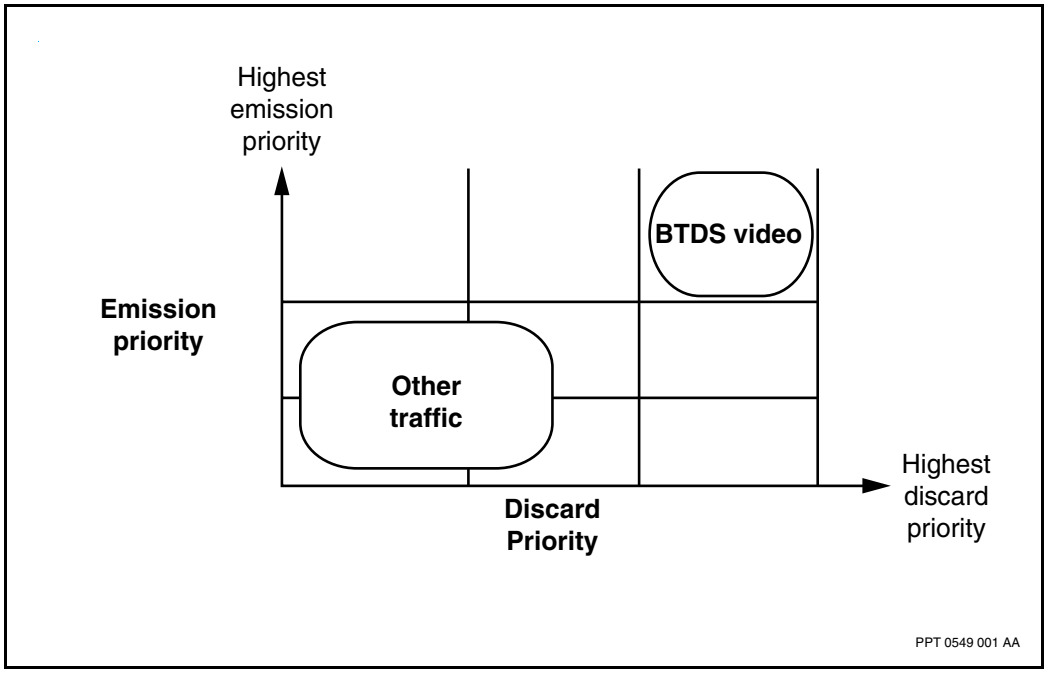


Figure 16
Emission and discard priority setup for BTDS video traffic



Other bandwidth considerations

When considering running BTDS over a pre-existing Nortel Networks Multiservice Switch trunk,

- determine the usual bandwidth used by the pre-existing traffic
- account for burstiness and control traffic
- estimate the amount of bandwidth needed by the BTDS traffic
- be sure that the combined bandwidth is available
- always try to run video over high-speed links
- always consider BTDS bandwidth to be in constant use

Chapter 4
Establishing connections—route selection

See the following section for information about route selection:

- “Selecting paths” (page 73)
- “Restricting traffic” (page 76)
- “Restricting paths” (page 78)
- “Provisionable path and related Trunk attributes” (page 82)

You can use the criteria in this section 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 your network resources.

Note 1: When you activate provisioning changes, service disruptions occur.

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

This section provides details on the following topics:

- “Minimization criteria—cost and delay” (page 74)

- "Specifying a maximum cost for a path" (page 76)
- "Specifying a maximum delay for a path" (page 76)

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. The *pathAttributeToMinimize* attribute under the *PLC* component is used to specify cost or delay.

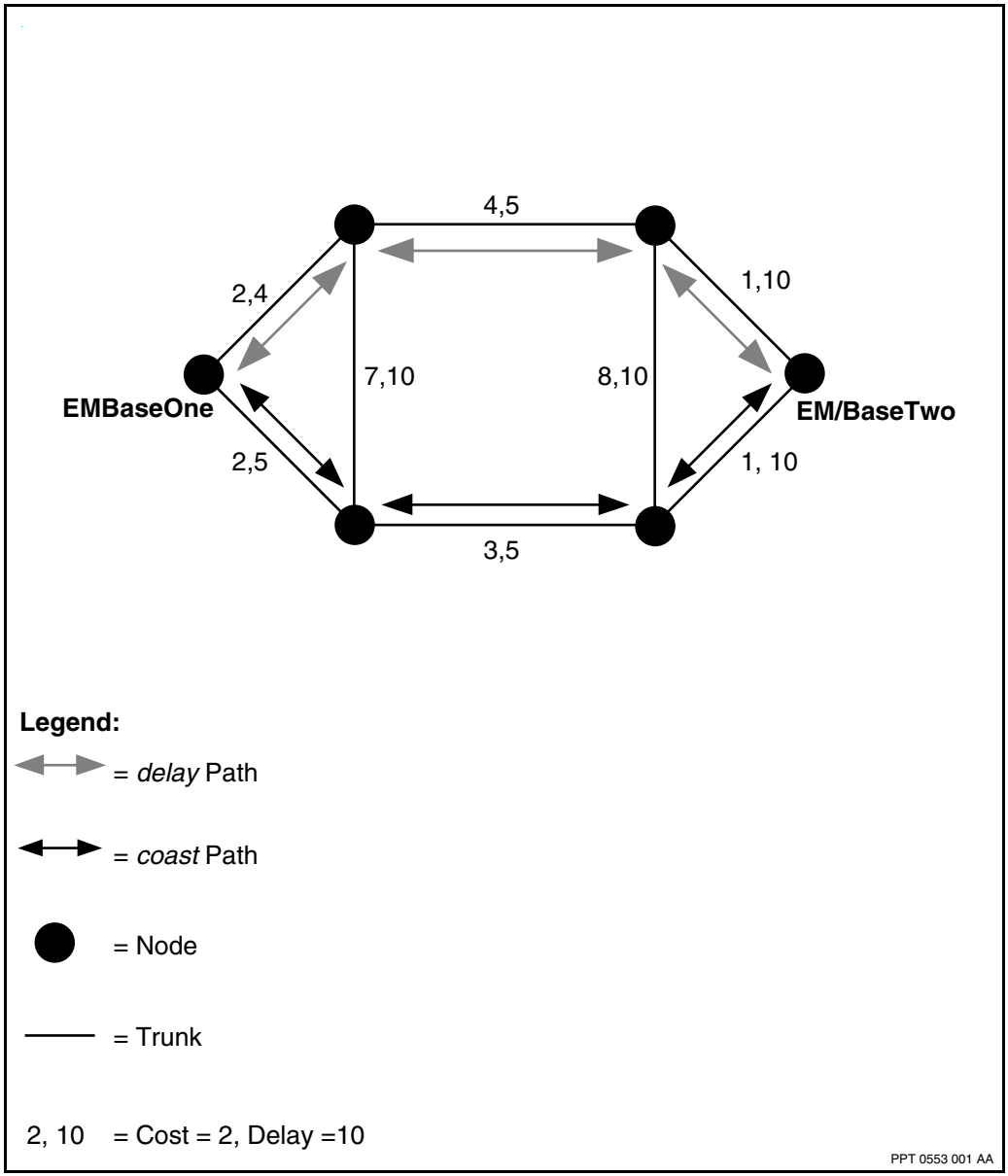
The routing system computes a minimum path from the values that you have assigned to the Nortel Networks Multiservice Switch trunks (cost) or from delay values that are associated with each Multiservice Switch trunk. This is shown in figure "Path for cost or delay using trunkAttributeToMinimize" (page 75).

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

Cost can be an actual monetary value or any unit that you want to use, although the same parameter must be used consistently across the network (for example you could not mix cost and some other parameter). 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 17
Path for cost or delay using trunkAttributeToMinimize



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 cost to mean any parameter. For example, this may be the geographic distance, hop count, or real dollar value.

The sum of the *trunkCost*-attribute values of all Multiservice Switch trunks used in the path will be less than or equal to the value specified by *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 Multiservice Switch 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 and voice).

Restricting traffic

This section contains information on the following topics:

- “Restricting certain types of traffic to specific trunks” (page 76)
- “Restricting traffic to certain types of trunks” (page 77)

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 Path Administrator* 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* to a trunk's *supportedTrafficTypes*. In other words, a service's *requiredTrafficType* must be included in a trunk's *supportedTrafficTypes* list or the trunk is not selected for the path. For example, if the service's *requiredTrafficType* is data, only Multiservice Switch trunks with *supportedTrafficTypes* that include data are selected for the path.

As another example, consider two services (HTDS/10 and BTDS/20) running on two Multiservice Switch trunks over ATM: 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* component, allows you to do this for up to eight different types of trunks.

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

Restricting paths

This section provides details on the following topics:

- "Security" (page 78)
- "Defining general parameters to restrict paths" (page 80)
- "Specifying a path manually" (page 81)

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 trunks, for example.

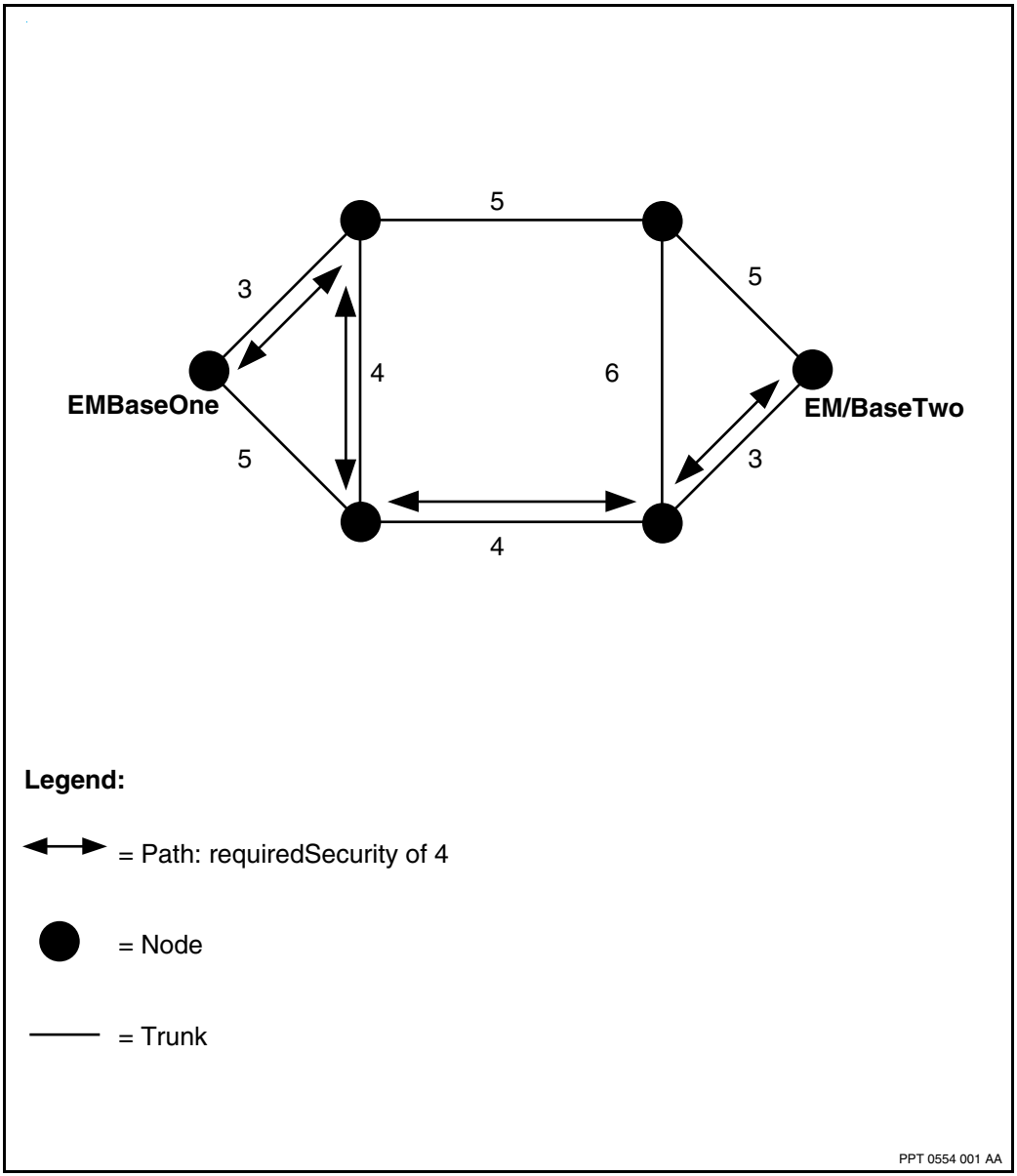
PORS has an option that allows you to specify the minimum security level of a path. To do this, provision a security value for the Multiservice Switch trunks in your Multiservice Switch network using the *trunkSecurity* attribute under the Trunk Path Administrator component. When you provision the connection,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

The connection will only use Multiservice Switch 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 requiredSecurity value of 4" (page 79) (a lower number 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 18
Path determined using a requiredSecurity value of 4



Defining general parameters to restrict paths

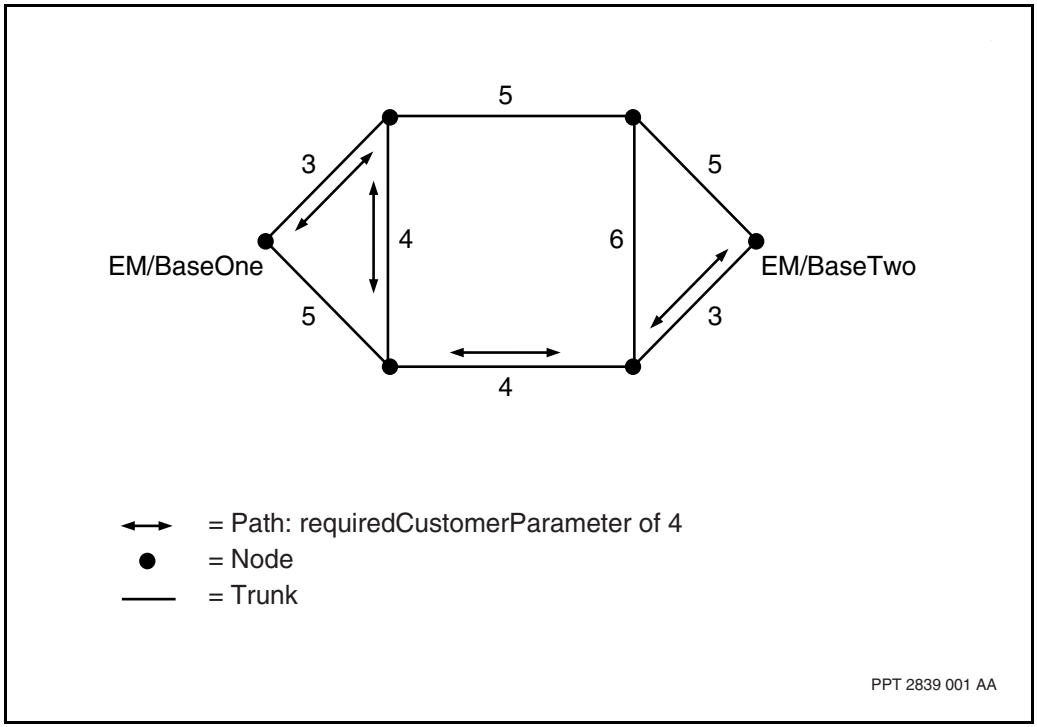
It may be convenient to be able to restrict certain classes of paths to certain Nortel Networks Multiservice Switch trunks. Most of the commonly used qualifiers are represented in security and traffic type. This is an additional option, to be used for any function that you deem appropriate.

PORS allows you to restrict certain paths to certain Multiservice Switch 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 Path Administrator* component. When the PLC 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 requiredCustomerParameter of 4" (page 81).

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

Figure 19
Path using a requiredCustomerParameter of 4



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 Multiservice Switch 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.

If you want to override the automatic selection of a path and specify the trunks manually, use the *manualPath* attribute of 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 PLC attributes, including bandwidth requirements.

Provisionable path and related Trunk attributes

Table "Provisionable path and related Trunk attributes" (page 82) summarizes information about the attributes discussed in this chapter.

Table 3
Provisionable path and related Trunk attributes

Path requirement attribute	Trunk attribute	Trunk is candidate for route if...
pathAttributeToMinimize (value = cost or delay)	**trunkCost, trunkDelay**	**trunkCost** less than **maximumAcceptableCost** **trunkDelay** less than **maximumAcceptableDelay**
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)	**requiredTrafficType** is element of **supportedTrafficTypes**
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)	**trunkType** is element of **permittedTrunkTypes**
requiredSecurity (value = number)	**trunkSecurity** (value = number)	**trunkSecurity** is less than or equal to **requiredSecurity**

(Sheet 1 of 2)

Table 3 (continued)
Provisionable path and related Trunk attributes

Path requirement attribute	Trunk attribute	Trunk is candidate for route if...
requiredCustomerParameter (value = number)	**trunkCustomerParameter** (value = number)	**trunkCustomerParameter** is less than or equal to **requiredCustomerParameter**
manualPath (value = list of trunk component IDs)	Trunk component name for example: EM/NODER9 trunk/202	trunk component name is one of manualPath

(Sheet 2 of 2)

Chapter 5
BTDS implementation guidelines

This section describes engineering aspects that affect the Bit Transparent Data Service (BTDS) and illustrates their effect on the operating characteristics of this service. The topics include

For information on BTDS connections between voice and MVP-E FPs, see "Voice and MVP-E function processor interworking restrictions" (page 91).

BTDS capabilities

BTDS supports

- up to eight Bit Transparent Data Service interfaces (ports) using V.35 function processor (FP) physical connections with speeds of 9.6 kbit/s up to 2 Mbit/s for each interface.
- up to eight Bit Transparent Data Service interfaces using V.11 FP physical connections with speeds of 9.6 kbit/s up to 2 Mbit/s for each interface.

  Note: There is a maximum combined throughput for the eight interfaces of the V.11 FP.

- up to four Bit Transparent Data Service interfaces using DS1, 8pDS1 or E1 FPs in single-link mode.

- up to eight Bit Transparent Data Service interfaces using DS1, 8pSD1 or E1 FPs in fractional mode. In fractional mode (ports 1 and 3 only), one primary rate interface can have one to four fractions associated with it. Also, there is a maximum combined throughput for the four interfaces of the DS1, 8pDS1 or E1 FP.
- for non-frame aligned BTDS, up to twenty-four Bit Transparent Data Service interfaces can be defined for a DS1 MVP-E FP and thirty-one for an E1 MVP-E FP.

For more information on function processors, see NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*. See the Release Notes for actual combinations of speeds currently supported.

BTDS interconnection information and requirements

To successfully establish a BTDS connection between two FPs, both ends must share compatible provisioning information. A BTDS connection attempt succeeds when the FPs determine that they share compatible provisioning data.

You cannot provision either the *Mcdn* or *Dpnss1* call discriminator sub-components on MVP-E FPs. As well, speeds at both ends of a BTDS connection must be the same.

Either internal or external synchronization may be used. With internal synchronization, the *clockingSource* attribute should be set to module for all types of FPs. For V.11 and V.35, the ports must be set to DCE. Internal synchronization facilitates fault tracing. If external synchronization is used, the clocking source should be set to *line* for DS1 and E1, and DS1 MVP-E and E1 MVP-E FPs. For V.11 and V.35 FPs, the ports must be set to DTE.

Source and destination function processors (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). Table “Supported BTDS interconnections” (page 90) describes the supported BTDS interconnections. The following sections contain requirements for all possible interconnections:

- “V.11 to V.11 or V.35 to V.35” (page 87)
- “DS1 or 8pDS1 to DS1or 8pDS1” (page 87)

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. For values of generated clock rates see NN10600-170 *Nortel Networks Multiservice Switch 7400 Hardware Description*. 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.

DS1 or 8pDS1 to DS1or 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 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 channel:

- Each channel can contain 1 to 24 timeslots in increments of one timeslot at a rate of 56 kbit/s or 64 kbit/s for each timeslot.

  Note: Applies to any combination of DS1 and/or 8pDS1 ports.

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 channel:

- Each 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

For these interconnections

- Both ends must be operating in Bit Transparent Data Service mode.
- 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

Both ends must be operating in Bit Transparent Data Service mode. As well, line status leads must be terminated locally.

V.11 or V.35 to voice or MVP-E FP

There is a limit of 1*64 kbit/s channel for a DS1/E1/TTC2M MVP-E FP.

DS1, 8pDS1 or E1 to DS1 or MVP-E FP

There is a limit of 1*64 kbit/s channel for a DS1 MVP-E or E1 MVP-E FP.

When you create a BTDS connection between a DS1 MVP-E FP and a DS1 FP, you must set the *zeroCoding* attribute to b8zs.

DS1 MVP-E to DS1 MVP-E

For non-frame aligned BTDS, there is a limit on the number of n*64 kbit/s channels, where n has the value of 1 for a DS1 MVP-E FP.

The DS1 MVP-E FP does not support frame-aligned BTDS

BTDS supports interworking connections between voice and MVP-E FPs. See “Voice and MVP-E function processor interworking restrictions” (page 91) for more information.

E1 MVP-E to E1 MVP-E

For non-frame aligned BTDS, there is a limit on the number of n*64 kbit/s channels, where n has the value of 1 for an E1 MVP-E FP.

The E1 MVP-E FP does not support frame-aligned BTDS.

BTDS supports interworking connections between voice and MVP-E FPs. See “Voice and MVP-E function processor interworking restrictions” (page 91) for more information.

TTC2M MVP-E to TTC2M MVP-E

For non-frame aligned BTDS, there is a limit on the number of n*64 kbit/s channels, where n has the value of 1 for a TTC2M MVP-E FP.

The TTC2M MVP-E FP does not support frame-aligned BTDS.

BTDS supports interworking connections between voice and MVP-E FPs. See “Voice and MVP-E function processor interworking restrictions” (page 91) for more information.

Table 4
Supported BTDS interconnections

	FP type							
FP type	E1	DS1	V.11	V.35	E1 MVP-E	DS1 MVP-E	TTC2M MVP-E	8p DS1
E1	X		X	X	X	X	X	
DS1		X	X	X	X	X	X	X
V.11	X	X	X	X	X	X	X	X
V.35	X	X	X	X	X	X	X	X
E1 MVP-E			X	X	X			X
DS1 MVP-E			X	X		X		X
TTC2M MVP-E			X	X			X	X
8p DS1		X	X	X	X	X	X	X

BTDS end-to-end negotiation requirements

When you attempt to establish a BTDS connection between two FPs running Nortel Networks Multiservice Switch R5.1 software, end-to-end negotiation occurs. During end-to-end negotiation, the source and destination FPs exchange provisioning information. A BTDS connection attempt succeeds when the FPs verify that they share compatible provisioning data.

In some cases, the end-to-end negotiation process causes an FP to modify specific provisioned values (see “Voice and MVP-E function processor interworking restrictions” (page 91)). If the end-to-end negotiation process reveals mismatched provisioning data, the connection request is rejected. The operational attribute *serviceFailureReason* contains the reason the connection failed. For example, if one end uses frame-aligned timeslots and the other end uses unaligned, the connection fails and *serviceFailureReason* attribute contains the compatibility error mismatchedTimeSlotAlignment. For more information about failed connection requests, see “Configuring the remoteName attribute” (page 91).

Configuring the remoteName attribute

The *Btds PermanentLogicalConnection* (*Plc*) component's *remoteName* attribute has an important role in the end-to-end negotiation process. If the *Btds Plc* component's *remoteName* attribute is set incorrectly (for example, because of improper syntax), a connection request is rejected even if the source and destination FPs share compatible provisioning data. Alternately, if you do not specify a value for the *remoteName* attribute, the *Btds Plc* component accepts connection requests from any remote *Btds Plc* component. However, with no value specified for the *remoteName* attribute, the *Btds Plc* component cannot originate a connection request and the connection establishment process is slower. See NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference* for more information about setting the *remoteName* attribute.

Voice and MVP-E function processor interworking restrictions

For BTDS connections between voice and multipurpose voice platform enhanced echo cancellation (MVP-E) function processors (FPs), interworking comprises specific BTDS features.

To establish BTDS connections between voice and MVP-E FPs, ensure that you

- set the *Btds* component's *timeSlotAlignment* attribute to unaligned on both FPs

  MVP-E FPs do not support frame-aligned BTDS. If you set the *timeSlotAlignment* attribute to frameAligned on a voice FP, the BTDS connection attempt fails during end-to-end negotiation. If the BTDS connection fails for this reason, the operational attribute *serviceFailureReason* contains the value mismatchedTimeSlotAlignment.

- do not provision either the *Mcdn* or *Dpnss1* call discriminator sub-components

  MVP-E FPs do not support the voice/data call discrimination feature. You cannot provision either *Mcdn* or *Dpnss1* call discriminator sub-components on an MVP-E FP.

  Also, end-to-end negotiation rejects a connection attempt between an MVP-E FP and a voice FP provisioned with one of the call discriminator sub-components. If the BTDS connection fails for this reason, the operational attribute *serviceFailureReason* contains the value mismatchedCallDiscrimination.

- establish connections with voice FPs running Nortel Networks Multiservice Switch R5.0 or greater software

  An MVP-E FP cannot establish a BTDS connection with a voice FP running Multiservice Switch pre-R5.0 software. As well, end-to-end negotiation only occurs between FPs—either MVP-E FPs, voice FPs, or both—running Multiservice Switch R5.1 software.

- install Multiservice Switch R5.1 software first before you reprovision the *cardType* attribute when migrating BTDS from voice to MVP-E FPs.

Chapter 6
Component monitoring

The following sections describe the component and attribute monitoring process as it relates to the Bit Transparent Data Service (BTDS):

Monitoring network clock synchronization

Only list and display apply to the *Network Sync* component. The OSI state will show one of two valid states:

- unlocked, enabled, and busy–the *Network Sync* component is currently synchronized to the *activeReference*, or there are no references currently provisioned and the component is free running.
- unlocked, disabled, and idle–the *Network Sync* component is currently in the process of tracking the *activeReference* and becoming synchronized with the network or none of the provisioned reference ports are valid. For example, the provisioned ports are disabled.

The *clockSyncState* will show three valid states: freeRun, synchronizing, and synchronized. These states are described in NN10600-550 *Nortel Networks Multiservice Switch 7400/15000/20000 Common Configuration Procedures*.

The *activeReference* shows which port is currently being tracked. The *standbyReference* shows the port that will be tracked if the *activeReference* fails.

BTDS operational attributes

The following examples display BTDS operational attribute information. Table “BTDS operational attributes” (page 96) lists BTDS operational attributes. Not included in the table is the *serviceFailureReason* operational attribute, found under each *Btds* component instance.

To display *Framer* component operational attributes, type the following command:

```
d Btds/<n> Framer
```

```
Btds/<n> Framer
   adminState             = unlocked
   operationalState       = enabled
   usageState             = busy
   frmFromIf              = 16891692
   suppressedFrames       = 0
   frmToIf                = 16891692
   lrcErrors              = 0
   frmLostInNetwork       = 0
   frmUnderRuns           = 0
   frmDumped              = 0
   igChanStatus           = normal
   egChanStatus           = normal
   igCtrlFrames           = 0
   egIdleFillFrameOut     = 0
   egQueueNumberOfFrames  = 4
   egQueueTime            = 30382 usec
   egIdleUnderRunCnt      = 0
   egNormalStateThrldCnt  = 0
   egIdleStateThrldCnt    = 0
   currentCellSize        = 44 bytes
```

To display *LogicalConnection* component operational attributes, type the following command:

d Btds/<n> Lco

```
Btds/<n> Lco
   state = pathUp
   path = "EM/BaseOne Trk/1 LCh/10"
"EM/BaseTwo testap/1"
   overrideRemoteName =
   end = calling
   costMetric = 128
   delayMetric = 4 ms
   roundTripDelay = 2 ms
   setupPriority = 2
   bumpPreference = bumpWhenNecessary
   holdingPriority = 2
   requiredTxBandwidth = 320000 bit/s
   requiredRxBandwidth = 320000 bit/s
   requiredTrafficType = data
   permittedTrunkTypes = terrestrial satellite tt1 tt2
```

```
tt3 ~tt4 ~tt5 ~tt6
   requiredSecurity = 4
   requiredCustomerParameter = 4
   emissionPriority = 1
   discardPriority = 2
   pathType = normal
   retryCount = 2
   pathFailureCount = 1
   reasonForNoRoute = none
   lastTearDownReason = insufficientTxBandwidth
   pathFailureAction = reRoutePath
   optimization = enabled
   pathUpDateTime = 1995-09-22 19:20:02.26
   pktsToNetwork = 15165382 packets
   bytesToNetwork = 454961460 bytes
   pksFromNetwork = 15162886 packets
   bytesFromNetwork = 454886580 bytes
```

Table 5
BTDS operational attributes

OSI state	BTDS framer	BTDS Lco	BTDS Mcdn or BTDS Dpnss1
adminState usageState	frmFromIf frmToIf	state 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.	timeSlotOpMode newVoiceCalls

(Sheet 1 of 3)

Table 5 (continued)
BTDS operational attributes

OSI state	BTDS framer	BTDS Lco	BTDS Mcdn or BTDS Dpnss1
proceduralStatus	lrcErrors	end	newDataCalls
alarmStatus	frmLostInNetwork	costMetric	voiceToData
unknownStatus	frmUnderRuns	delayMetric	dataToVoice
operationalState	frmDumped	roundTripDelay	callClears
availabilityStatus	adminState	setupPriority	frmProcessed
controlStatus	usageState	holdingPriority	frmInvalid
standbyStatus	operationalState	requiredTxBandwidth	frmFromIf
framesToIf	suppressed Frames	requiredRxBandwidth	aborts
framesFromIf	igChanStatus	requiredTrafficType	crcErrors
discarded FramesToIf	egChanStatus	permittedTrunkTypes	nonOctetErrors
discarded FramesFromIf	igCtrlFrames	requiredSecurity	overruns
	egIdleFillFrame Cnt	requiredCustomerParameter	largeFrmErrors
	egQueueNumber OfFrames	emissionPriority	
	egQueueTime	discardPriority	
	egIdleUnderRun Cnt	pathType	
	egNormalStateThr ldCnt	retryCount	
	egIdleStateThrld Cnt	pathFailureCount	
	currentCellSize	lastTearDownReason	

(Sheet 2 of 3)

Table 5 (continued)
BTDS operational attributes

OSI state	BTDS framer	BTDS Lco	BTDS Mcdn or BTDS Dpnss1
		overrideRemoteName bumpPreference reasonForNoRoute pathFailureAction optimization pathUpDateTime pktsToNetwork bytesToNetwork pktsFromNetwork bytesFromNetwork	
(Sheet 3 of 3)			

BTDS related Trunk operational attributes

The following examples display BTDS-related *Trunk* component operational attributes. Table "BTDS-related Trunk and PathAdministrator attributes" (page 100) lists BTDS *Trunk* and *PathAdministrator* component operational attributes.

To display *Trunk* component operational attributes, type the following command:

d Trk/<n>

```
Trk/<n>
   adminState              = unlocked
   operationalState        = enabled
   usageState              = busy
   availabilityStatus      =
   proceduralStatus        =
   controlStatus           =
   alarmStatus             =
   standbyStatus           = notSet
   unknownStatus           = false
   remoteComponentName     = EM/BASEONE TRUNK/0027
   measuredSpeedToIf       = 1918000 bit/s
   measuredRoundTripDelay = 2 ms
   maxTxUnit               = 32768 byte
   areYouThereModeEntries = 0
   pktFromIf               = 1465
   trunkPktFromIf          = 1189
   trunkPktToIf            = 1190
   discardUnforward        = 1
   discardTrunkPktFromIf   = 0
   intPktFromIf            = 437087
   discardIntUnforward     = 0
   stagingAttempts         = 1
```

To display *PathAdministrator* component operational attributes, type the following command:

d Trk/<n> pa

```
Trk/<n> Pa
   state = up
   usedLc = 1
   negotiatedMaxLc     = 4096
   maxReservableBwOut = 640000 bit/s
   overReservedBwOut   = 0 bit/s
   unreservedBwOut     = 320000 bit/s
   reservedBwOutByHp   = 2 : 320000 bit/s
   attemptCountByHp    = 2 : 10
   failCountByHp       =
```

```
clearCountByHp      = 2 : 9
bumpCountByHp =
clashCount     = 0
```

Table 6
BTDS-related Trunk and PathAdministrator attributes

Trunk	PathAdministrator
adminState	state
operationalState	usedLc
usageState	negotiatedMaxLc
availabilityStatus	maxReservableBwOut
proceduralStatus	overReservedBwOut
controlStatus	unreservedBwOut
alarmStatus	reservedBwOutByHp
standbyStatus	attemptCountByHp
unknownStatus	failCountByHp
remoteComponentName	clearCountByHp
measuredSpeedToIf	bumpCountByHp
measuredRoundTripDelay	clashCount
maxTxUnit	
pktFromIf	
discardUnforward	
intPktFromIf	
discardIntUnforward	
areYouThereModeEntries	
trunkPktFromIf	
trunkPktToIf	
discardTrunkPktFromIf	
stagingAttempts	

Chapter 7
Troubleshooting

The following sections provide information and guidelines on what steps you can take to solve problems that may occur after you have installed BTDS. Also included are provisioning checklists, which help ensure a successful BTDS setup. See the following sections for more information:

Provisioning checklists

Refer to the provisioning checklists in the following sections to avoid having problems establishing a BTDS connection:

Installation

Make sure you have done the following:

- Have you installed PORS and BTDS according to the instructions in NN10600-270 *Nortel Networks Multiservice Switch 7400/15000/20000 Software Installation*?

- Do you have the latest software on every Nortel Networks Multiservice Switch 7400 node?
- Have you added Trunk pathAdmin to every Multiservice Switch 7400 node that could be a path candidate?

Network clock synchronization

Make sure you have done the following:

- Have you identified a master clock reference?
- Have you planned and established a tree structure of nodes; each synchronizing their clock signal with the last, branching back to the master reference?
- Have you checked for synchronization loops?

Provisioning BTDS 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 checked your spelling?
- Have you used the check, save, activate, and confirm commands?
- Have you provisioned both ends of the connection?

Specifying attribute values

Make sure you have done the following:

- Are you sure that the bandwidth is available? (Remember that you are probably sharing the total bandwidth with connectionless routing.)
- Are the attributes under the *Trunk* component provisioned identically at both ends of the connection? If they are not, you will not get a connection.

- Several attributes have provisioning dependencies. See NN10600-060 *Nortel Networks Multiservice Switch 7400/15000/20000 Component Reference* for detailed BTDS component and attribute descriptions and available values.

  Note: For BTDS connection between voice and MVP-E FPs, see "Voice and MVP-E function processor interworking restrictions" (page 91).

  Note: With the exception of the *remoteName* attributes, the attributes at both ends of a BTDS connection must contain identical provisioning information.

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 Nortel Networks Multiservice Switch 7400 fault management, they are described separately in 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

BTDS-related alarms

The alarms related to BTDS are as follows:

- 7018 0001 to 7018 0004 Path Administrator-related alarms
- 7018 1001 and 7018 1002 LCo-related alarms
- 7019 0001 BTDS-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 101) 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 flowchart in figure “Flowchart: an example of troubleshooting using LCo” (page 105) may help you to determine the answers to these questions. For component and attribute monitoring information, refer to “Component monitoring” (page 93).

Figure 20
Flowchart: an example of troubleshooting using LCo

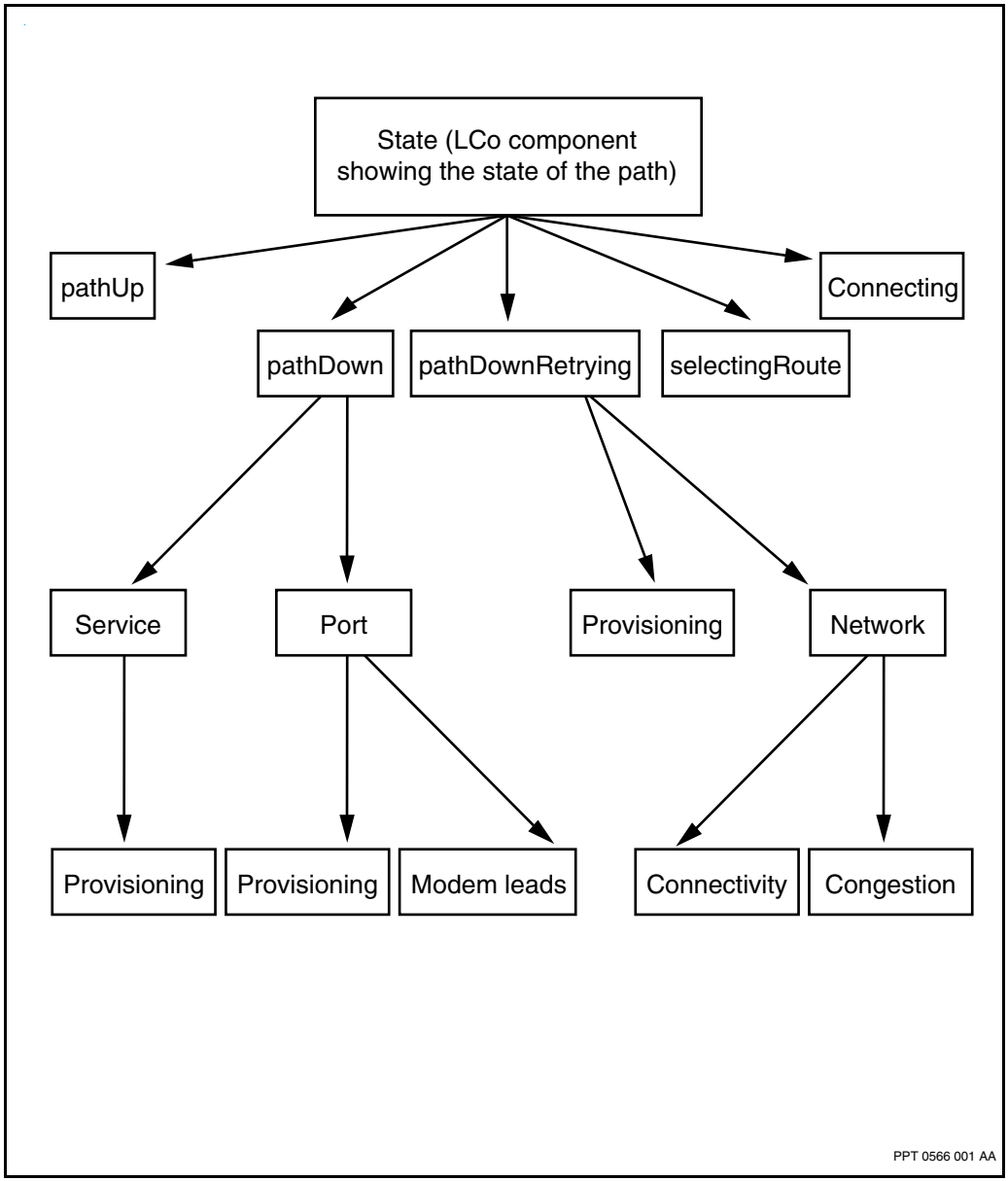


Table 7
Handling problems

Problems that may occur	Probable causes	Corrective measures
Unable to provision BTDS.	Error in card provisioning.	Check the card. If an orange light is glowing, check the *logicalProcessor* and *logicalProcessorType* attribute settings. Reprovision using correct lp and lpt information. 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 7 (continued)
Handling problems

Problems that may occur	Probable causes	Corrective measures
The connection doesn't come up.	Hardware connections.	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 101). Trunk provisioning information must be identical for both ends and consistent with the PLC provisioning of the connection. Reprovision with correct information. If a BTDS connection does not come up, check the *Btds* component's *seviceFailureReason* attribute. For connections between a voice and MVP-E FP, see also "Voice and MVP-E function processor interworking restrictions" (page 91) for more information.
(Sheet 2 of 4)		

Table 7 (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 Nortel Networks Multiservice Switch trunks in the path. Reprovision using less bandwidth if it is not needed or re-engineer the network to make bandwidth available.
	If *manualPath* attribute has been used, one of the nodes or trunks used may have failed.	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 NN10600-550 *Nortel Networks Multiservice Switch 7400/ 15000/20000 Common Configuration Procedures* for provisioning information.
Note: For recurring problems 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.
Connection goes down and does not reset.	Under extreme circumstances (such as there being no suitable trunk available) a path may take 1–2 minutes to reroute.	Wait 1–2 minutes and check to see if rerouting has occurred.

(Sheet 3 of 4)

Table 7 (continued)
Handling problems

Problems that may occur	Probable causes	Corrective measures
Connection goes down and does not reset (Continued).	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.	Network clock synchronization is not setup properly.	Check that all of the trunks in the path are properly provisioned with network clock synchronization.
	Congestion.	Check bandwidth utilization. Take steps to reduce congestion if the problem continues.
	Poor trunk-error performance.	See NN10600-435 *Nortel Networks Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*.
Network clock synchronization remains in *coarseAquisition* for more than one minute.	Improper provisioning of the master.	Network clock synchronization may be provisioned for local where module is the proper value. Check and reprovision with correct value.
	Improper provisioning of references.	Carefully verify provisioning of references.
		As a last resort, quickly lock and unlock the port to which network clock synchronization is attempting to sync (where it is in coarse acquisition)
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.
The BTDS framer operational attribute entitled “frmUnderRuns” is increasing in value and the attribute “frmLostInNetwork” value is not changing.	The cross network delay is varying.	Increase the value for the BTDS framer attribute “insertedOuputDelay”.

(Sheet 4 of 4)

Troubleshooting examples

The following examples represent two possible problems which may occur. The problem in Example 1 represents a route that has failed to come up. The problem in Example 2 represent video equipment connection problems and video picture quality problems.

Example 1

The operator has provisioned a route that has failed to come up. The operator looks at the *reasonForNoRoute* attribute to determine the reason.

```
d btds/<n> lc reasonForNoRoute

reasonForNoRoute          = unknownRemoteNodeName
remoteName                = /NodeR2b
```

The *reasonForNoRoute* attribute indicates that the *remoteName* attribute is unknown. In this case the value of the *remoteName* attribute is also displayed and improper syntax was used when specifying the name of the remote Nortel Networks Multiservice Switch node.

Example 2

An operator finds the following video problem. The logical connection is up on a BTDS connection but the local video equipment cannot connect with the far-end, or the video picture seems to freeze every few seconds.

To locate the problem, the operator can perform the following steps to display the BTDS *Framer* component at both ends of the connection.

```
d btds/601 framer

Btds/601 Framer
   adminState = unlocked
   operationalState = enabled
   usageState       = busy
   frmFromIf        = 5905803
   suppressedFrames = 0
   frmToIf          = 5905803
   lrcErrors        = 0
   frmLostInNetwork = 0
   frmUnderRuns     = 400
   frmDumped        = 0
```

```
   igChanStatus = normal
   egChanStatus = normal
   igCtrlFrames = 0
   egIdleFillFrameCnt = 0
   egQueueNumberOfFrames = 3
   egQueueTime = 30382 usec
   egIdleUnderRunCnt = 0
   egNormalStateThrldCnt = 0
   egIdleStateThrldCnt = 0
   currentCellSize = 44 bytes
```

d Btds/604 framer

```
Btds/604 Framer
   adminState = unlocked
   operationalState = enabled
   usageState       = busy
   frmFromIf        = 5929605
   suppressedFrames = 0
   frmToIf          = 5927677
   lrcErrors        = 0
   frmLostInNetwork = 0
   frmUnderRuns     = 0
   frmDumped        = 400
   igChanStatus = normal
   egChanStatus = normal
   igCtrlFrames = 0
   egIdleFillFrameCnt = 0
   egQueueNumberOfFrames = 3
   egQueueTime = 30382 usec
   egIdleUnderRunCnt = 0
   egNormalStateThrldCnt = 0
   egIdleStateThrldCnt = 0
   currentCellSize = 44 bytes
```

If the *frmUnderRuns* and *frmDumped* attribute counts are increasing on both ends of the connection, then the cause of the video problem is a result of the line speeds at either end of the connection not being synchronized. To find the problem:

- Check that the clock rates at either end of the connection are the same.
- If network clock synchronization is being used:

— Check the network synchronization tree to see if it is setup properly.

— Check for any outstanding network synchronization alarms.

- If external synchronization is being used:

  — Check the external network synchronization tree.

  — If a DS1 or E1, or DS1, E1 or TTC2M MVP-E FP is being used, the *clockingSource* attribute must be set to line.

  — If V.35 or V.11 FPs (or ports) are being used, the *linkMode* attribute on the end point FPs must be set to DTE.

Display the BTDS *Framer* component at both ends of the connection.

```
d btds/601 framer

Btds/601 Framer
adminState = unlocked
   operationalState = enabled
   usageState       = busy
   frmFromIf        = 5905803
   frmToIf          = 5905782
   lrcErrors        = 0
   frmLostInNetwork = 973
   frmUnderRuns     = 0
   frmDumped        = 0

d Btds/604 framer

Btds/604 Framer
   adminState = unlocked
   operationalState = enabled
   usageState       = busy
   frmFromIf        = 5929605
   frmToIf          = 5927677
   lrcErrors        = 0
   frmLostInNetwork = 489
   frmUnderRuns     = 0
   frmDumped        = 0
```

If the *frmLostInNetwork* operational attribute is increasing, then check the following:

- Are any of the Nortel Networks Multiservice Switch trunks that these connections go over in a congestion state? If yes, then the problem may be solved by re-engineering the trunks or by changing some of the PLC parameters for this service.
- Are the trunk facilities operating within specification? If no, then correct facility problems.

Display the E1 or DS1 component, if a voice or MVP-E FP is being used.

```
d lp/5 e1/0

Lp/5 E1/0
   adminState = unlocked
   operationalState = enabled
   usageState       = busy
   availabilityStatus =
   proceduralStatus   =
   controlStatus      =
   alarmStatus        =
   standbyStatus      = notSet
   unknownStatus      = false
   losAlarm           = Off
   rxAisAlarm         = Off
   lofAlarm           = Off
   rxRaiAlarm         = Off
   txAisAlarm         = Off
   txRaiAlarm         = Off
   multifrmLofAlarm   = Off
   rxMultifrmRaiAlarm = Off
   txMultifrmRaiAlarm = Off
   runningTime        = 12259
   errorFreeSec       = 12259
   erroredSec         = 0
   sevErroredSec      = 0
   sevErroredFrmSec   = 0
   unavailSec         = 0
   bpvErrors          = 0
   crcErrors          = 0
```

```
frmErrors           = 0
losStateChanges     = 0
slipErrors            = 15
```

If the operational attribute *slipErrors* is increasing, then there is a clock synchronization problem between Multiservice Switch 7400 and the external equipment.

Nortel Networks Multiservice Switch 7400

Operations: Bit Transparent Data Service

Release 6.1

Publication: NN10600-775
Document status: Standard
Document version: 6.1S1
Document date: August 2004
Printed in Canada

NORTEL NETWORKS