Nortel Multiservice Switch 7400/15000/20000

ATM Routing and Signalling Fundamentals

NN10600-702

Document status: Standard
Document issue: 7.2S1
Document date: March 2006
Product release: PCR7.2 and up
Job function: Product Fundamentals
Type: NTP
Language type: U.S. English

Contents

What's new

The following feature was added to this document:

- Load Re-Balancing on Parallel Links (page 10)

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

Load Re-Balancing on Parallel Links

The following section was added for this feature:

- Load Re-Balancing on Parallel Links (page 133)

Overview of ATM ports, interfaces, and connections

Use the following sections to learn more about ATM ports, interfaces, and connections.

Navigation

Physical ports

Ports are the physical interface between two Nortel Multiservice Switch nodes or between a Multiservice Switch node and external ATM devices. Ports directly correspond to the facility over which the network carries a Multiservice Switch trunk. Ports and the associated facility are the physical medium over which network nodes transmit and receive ATM cells.

For more information on physical ports and the supported interface standards for Multiservice Switch device function processors, see NN10600-170 *Nortel Multiservice Switch 7400 Hardware Description* and NN10600-120 *Nortel Multiservice Switch 15000/20000 Hardware Description*.

ATM interfaces

The ATM interface is the software counterpart to the physical port on the function processor. Through configuration, you establish the direct association between the interface and its physical port and, by extension, the physical link that is hard wired to that port. Each physical ATM link or IMA link group has one ATM interface at each end. The interface, and the protocol associated with that interface, govern all connections on the ATM link.

An ATM node can be one of the following:

- equipment that functions as an ATM end point
- an ATM network node, either within the same network or within an external network

Nortel Multiservice Switch supports four ATM interface types:

- Basic interface (page 12)
- User-to-network interface (page 12)
- Interim inter-switch signaling interface (page 12)
- ATM inter-network interface (page 12)
- Private network-to-network interface (page 13)

For requirements for configuring ATM interface characteristics, see ATM interface configuration (page 13).

Basic interface

A basic ATM interface does not support a signaling and routing protocol, nor UNI, IISP, AINI, and PNNI. Due to these characteristic, a basic interface supports only permanent connections.

For more information on PVCs, refer to Permanent virtual connections and paths (page 18).

User-to-network interface

The UNI supports the interaction between ATM terminal equipment and the network. The signaling protocol used on this type of interface is the ATM Forum UNI version 3.0, UNI Version 3.1 or UNI Version 4.0. UNI is based on ITU-T Q.2931. UNI interfaces also use integrated local management interface (ILMI) control procedures for dynamic address registration across the interface.

For more information on UNIs and ILMI address registration, see Address registration through ILMI (page 44).

Interim inter-switch signaling interface

The IISP interface supports the interaction between network nodes. The IISP signaling protocol on these interfaces supports switched connections between Nortel Multiservice Switch nodes and other Nortel Networks devices. IISP is based on the ATM Forum standard *Interim Inter-switch Signaling Protocol (IISP) Specification Version 1.0*. IISP also supports static routing.

ATM inter-network interface

AINI supports the interaction between network nodes. The AINI signaling protocol on these interfaces supports switched connections between Nortel Multiservice Switch nodes, other Nortel Networks devices, and devices from

other vendors. AINI is based on PNNI 1.0 signaling and is compliant with the ATM Forum standard ATM Inter-Network Interface Specification (af-cs-0125.000). Compared to IISP, AINI provides:

- enhanced connection recovery with standard crankback
- standard SPVC and SPVP services
- Quality of Service routing

Private network-to-network interface

The Nortel Multiservice Switch PNNI implementation supports PVC, SPVCs, and SVCs for the following applications:

- PNNI supports the permanent connection functionality, such that permanent ATM bearer connections and Multiservice Switch trunks over ATM can traverse PNNI interfaces.
- PNNI supports standards-based SPVPs and SPVCs between Multiservice Switch nodes and any ATM Forum PNNI Version 1.0 compliant vendor equipment that implements SPVPs and SPVCs.
- PNNI supports both point-to-point and point-to-multipoint SVC connections.

PNNI provides dynamic routing and quality of service (QoS) support based on the ATM Forum PNNI Version 1.0 standard. Network engineers can configure PNNI-based ATM switches to monitor network topology and available resources. With this configuration, the network automatically routes calls around points of congestion and failure.

With the rerouting capability, PNNI also allows established SVC, SVP, SPVC, and SPVP connections to recover automatically from network failures or move to better PNNI routes.

ATM interface configuration

To deploy the appropriate ATM interfaces on Nortel Multiservice Switch, configure the interface characteristics:

- OAM segment boundary

  Determine if the ATM interface resides on an operations, administration, and maintenance (OAM) segment boundary.

- traffic management

  The network administrator determines how to configure traffic management controls for an interface or connection, where these controls ensure that the network supports the traffic requirements of end stations.

  — route management

  — emission priority, queuing, and traffic scheduling

  - discard priority
  - packet-wise discard
  - connection admission control (CAC)
  - traffic shaping
  - traffic policing through usage parameter control (UPC)

- connections

  Configure the appropriate connection type for the ATM interfaces. For more information, see ATM connections (page 14).

- ATM virtual interfaces

  Multiservice Switch nodes can support multiple virtual interfaces on one physical port by tunnelling SVCs through permanent virtual paths (PVP). A virtual interface may be a UNI, IISP, AINI or PNNI interface.

  Each port requires one physical path to the core network. Through configuration, you can associate each virtual interface with a PVP, and each virtual interface has a dedicated signaling channel for setting up SVCs within the PVP. In this way, Multiservice Switch nodes can provide the required SVC tunnelling through a PVP network. For an application example, see Permanent paths that support virtual interfaces (page 197).

Attention: ILMI is not available for virtual UNIs.

ATM connections

The network administrator determines how many virtual channel connections (VCC) and virtual path connections (VPC) the ATM interface supports and which connection instance values these connections can have. The instance value of a VCC defines the VPI.VCI value for the channel connection (for example, VCC/16.32 defines a channel connection with a VPI of 16 and a VCI of 32). The instance value of a VPC defines the VPI value for the path connection (for example, VPC/120 defines a path with a VPI of 120).

For 2- and 3-port CQC-based function processors, the selection of valid connection instance values is possible by configuring the connection map for the ATM interface. For 8-port CQC-based function processors and ATM IP function processors, connection mapping is automatic. For more information, see ATM connection mapping for ATM IP function processors (page 32), Connection mapping for 8-port CQC-based function processors (page 33), and Connection mapping for APC-based function processors and 2- and 3-port CQC-based function processors (page 33)

A VCC can be either configured or dynamic. However, a VPC can be configured only. Use configured connections for permanent virtual connections (PVC). Use dynamic connections for switched virtual circuits (SVC). To establish an SPVC or SPVP, configure the connection at the source ATM interface. The network dynamically establishes the connection on call setup.

On Nortel Multiservice Switch, SVCs, SPVCs, and SPVP are supported over PNNI, UNI, AINI and IISP ATM interfaces. The basic type of ATM interface supports PVCs only.

For more information on ATM connections, see ATM connections (page 16).

ATM connections

Use the following sections to learn more about ATM connections infrastructure for ATM networking.

Navigation

Overview to ATM connection types

The ATM connections layer in Nortel Multiservice Switch device's ATM infrastructure allows the service provider to define, configure, and monitor ATM connections. In addition, it associates connections with the ATM services in the next higher infrastructure layer. The table ATM connection types: support by function processor (page 17) summarizes the connection types that Multiservice Switch nodes support:

ATM connection types: support by function processor

Connection type	End station to end station implementation		
	Point-to-point	Point-to-multipoint	
		Multiservice Switch 15000 and Multiservice Switch 20000	Multiservice Switch 7400
PVC	Supported on all Multiservice Switch node function processors providing ATM capability.	• 12-port E3 Atm • 4-port OC-3 Mm Atm • 4-port OC-3 SmIr Atm • 16-port OC-3 SmIr Atm • 4-port OC-12 SmIr Atm a • 1-port OC-48 Ch SmIr Atm a • 16-port OC-3 Pos Atm	• 32-port DS1Msa • 32-port E1Msa • 3-port DS3 Atm 2 • 3-port E3 Atm 2 • 2-port OC-3 Mm Atm 2 • 2-port OC-3 Sm Atm 2
SPVC	Supported on all Multiservice Switch node function processors providing ATM capability.	• 12-port E3 Atm • 4-port OC-3 Mm Atm • 4-port OC-3 SmIr Atm • 16-port OC-3 SmIr Atm • 4-port OC-12 SmIr Atm a • 1-port OC-48 Ch SmIr Atm a • 16-port OC-3 Pos Atm	• 32-port DS1Msa • 32-port E1Msa • 3-port DS3 Atm 2 • 3-port E3 Atm 2 • 2-port OC-3 Mm Atm 2 • 2-port OC-3 Sm Atm 2
SVC	Supported on all Multiservice Switch node function processors providing ATM capability.	Supported on all Multiservice Switch node function processors providing ATM capability.	
PVP	Supported on all Multiservice Switch node function processors providing ATM capability.	• 12-port E3 Atm • 4-port OC-3 Mm Atm • 4-port OC-3 SmIr Atm • 16-port OC-3 SmIr Atm • 4-port OC-12 SmIr Atm a • 1-port OC-48 Ch SmIr Atm a • 16-port OC-3 Pos Atm	• 32-port DS1Msa • 32-port E1Msa • 3-port DS3 Atm 2 • 3-port E3 Atm 2 • 2-port OC-3 Mm Atm 2 • 2-port OC-3 Sm Atm 2

(1 of 2)

ATM connection types: support by function processor (continued)

Connection type	End station to end station implementation		
	Point-to-point	Point-to-multipoint	
		Multiservice Switch 15000 and Multiservice Switch 20000	Multiservice Switch 7400
SPVP	Supported on all Multiservice Switch node function processors providing ATM capability.	Not supported.	
SVP	Supported on all Multiservice Switch node function processors providing ATM capability.	Not supported.	
The end station to end station implementation refers to the combination of source and destination points. For any connection, the end station may be a single station (point) or multiple stations (multipoint).			
(2 of 2)			

Permanent virtual connections and paths

ATM PVCs are virtual circuit data paths that run through the ATM network and consist of a combination of VCC segments and VPC segments. ATM VCCs are connections between two ATM network entities. ATM VPCs identify, or manage, groups of VCCs.

For more information on connections and paths, see the following sections:

- Connection elements (page 18)
- Overview to connection points (page 19)
- Connection end point (page 20)
- Segment end point (page 21)
- Connecting point (page 22)
- ATM connection configuration (page 22)

See also Hitless ATM services for Multiservice Switch nodes (page 28) for further details.

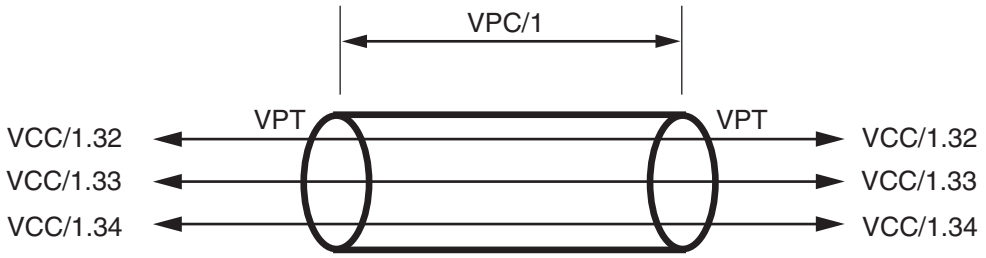
Connection elements

The figure Connection elements (page 19) shows a VPC bundling a number of associated VCCs. The illustration identifies the VPC, the VCCs, and the virtual path terminators (VPT). The virtual path (VP) layer is identified by its virtual path identifier (VPI). In this example, VPI = 1. Virtual circuit (VC) layer

cells are identified by their virtual channel identifiers (VCI). In this example, VCI = 32, 33, or 34. The VCI distinguishes one VC cell from another when the VC cells are multiplexed into a single VPC.

The VPT is the connection end point at each end of the VPC. The VPT is located at the point at which the hardware multiplexes and demultiplexes VC-layer cells from the VP layer. VPTs provide an efficient method of VP-layer fault management through termination of VPCs. Network management is easier using VPCs, since the service provider can implement and monitor a large number of VCCs through a much smaller number of VPCs.

Connection elements



Note: The VCI range of 0-31 is reserved for use by future standards

PPT 2626 001 AA

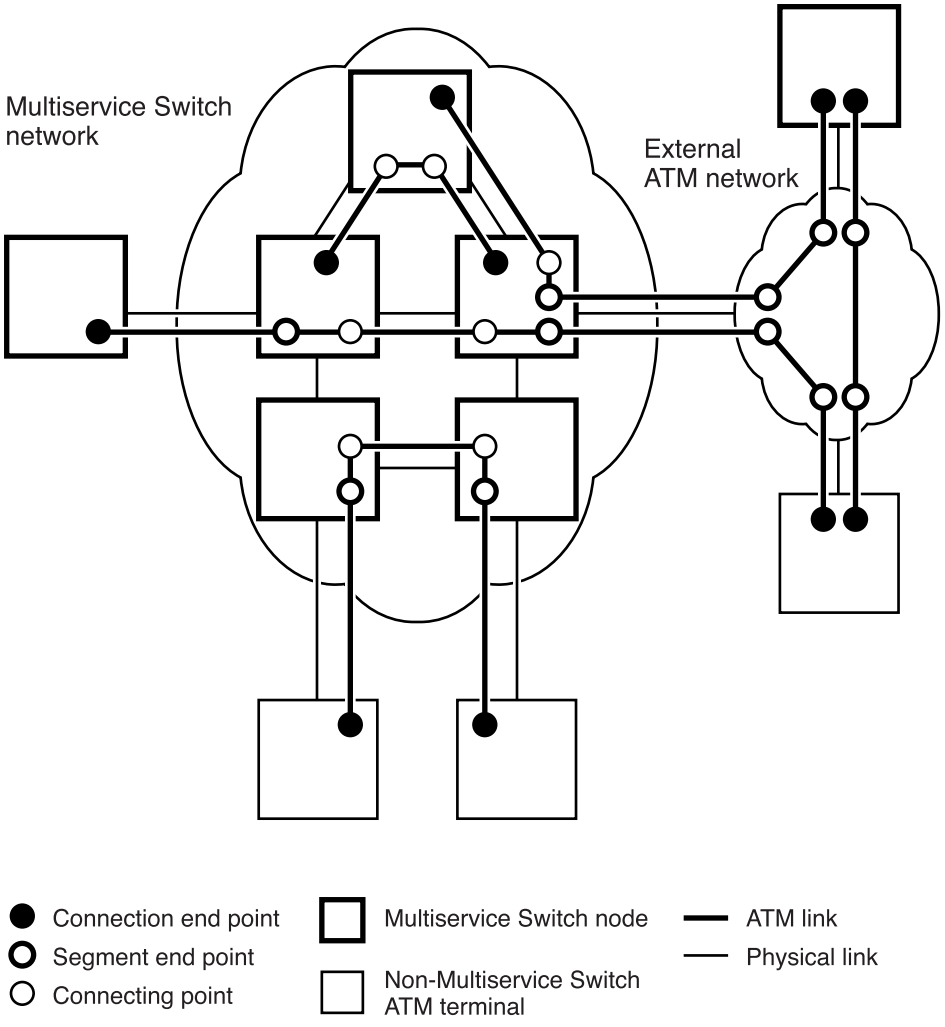
Overview to connection points

Each connection originates and terminates at a connection point. There are three types:

- connection end point
- segment end point
- connecting point

The figure Example of connection configurations (page 20) illustrates a network configuration that incorporates all three types.

Example of connection configurations



Connection end point

A connection end point terminates a connection at the ATM interface. At connection end points, the end station does the following:

- sources user traffic from upper layers through an AAL layer
- forwards user traffic to upper layers through an AAL layer
- terminates end-to-end OAM cells

Connections can terminate at any ATM interface.

The figures Example of connection configurations (page 20) and Typical ATM network with segment end points (page 22) illustrate the location of connection end points at the ATM terminals.

Segment end point

The network can manage an unswitched ATM bearer connection as one or more contiguous ATM segments. Each Nortel Multiservice Switch ATM connection segment is composed of contiguous Multiservice Switch modules connected through ATM physical links. Nodes along the connection send a segment OAM cell from one segment end point to the other. Segment OAM cells do not pass segment end points.

The figure Typical ATM network with segment end points (page 22) shows the location of segment end points.

A segment end point functions as a boundary between ATM OAM segments. A node does not relay the segment OAM cells on one side of the connection point (for example the link side) to the other side of the connection point (the node side). The segment OAM cells that are generated and transmitted to each side of the segment end point are processed independently at the level of the local ATM OAM segments. See the figure Typical ATM network with segment end points (page 22).

When you set up for a loopback, make sure both segment end points are of the same type (either VCC or VPC).

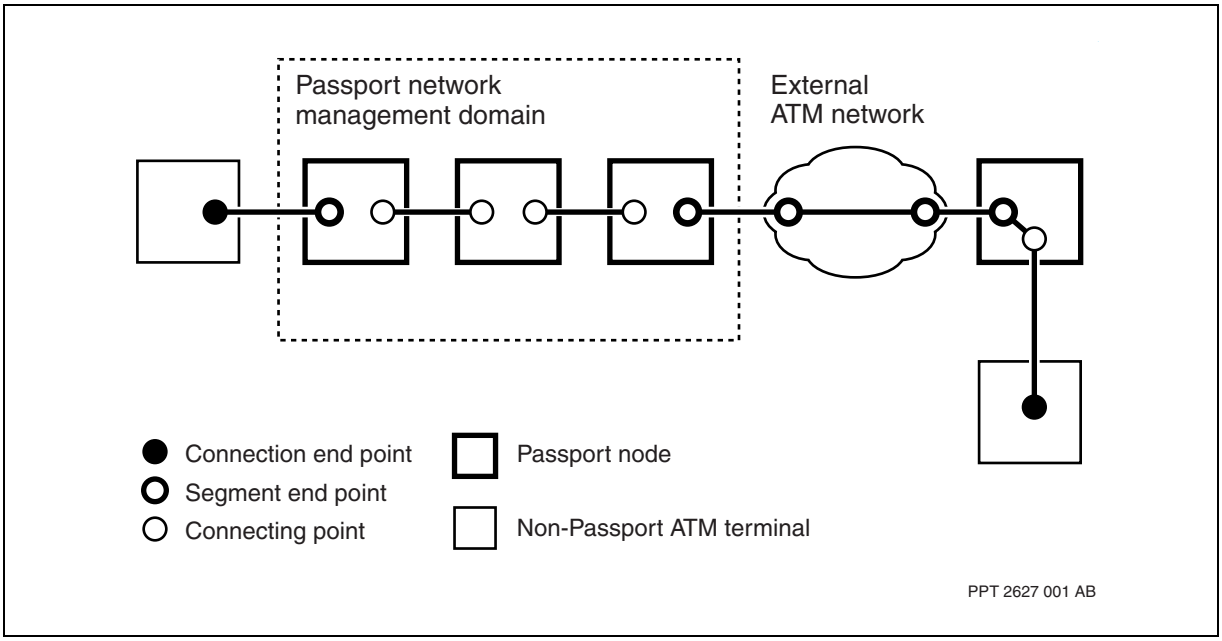
An ATM connection has OAM segments for two reasons:

- nodes (and therefore networks) monitor connectivity on the basis of the segment
- nodes complete connectivity tracing on the basis of the segment

Nodes monitor each segment to determine if end-to-end connectivity within the segment exists. Nodes monitor connectivity status by periodically generating OAM loopback cells from each segment end point.

When connectivity for a particular Multiservice Switch segment is lost, the network operator can query which modules in that segment are reachable along the connection path. This is useful when trying to isolate the source of the problem. OAM loopback cells are also used for this purpose.

Typical ATM network with segment end points



Connecting point

A connecting point relays end-to-end and segment OAM cells from one side of the connection point to the other.

Connecting points are configurable under any ATM interface independently of the OAM segment boundary definition for that interface. This independent configuration is done through the OAM segment boundary for a nailed-up relay point.

Do not configure a connection point with an OAM segment boundary value of *no* under an interface that is configured as an OAM segment boundary. This practice ensures that proprietary trace OAM cells and segment-related OAM cells are contained within the Nortel Multiservice Switch network.

ATM connection configuration

The characteristics of a PVC are based on the following configurable parameters:

- traffic contract
- a set of VCC and VPC traffic management strategies
- a set of VCC and VPC fault management strategies

The traffic descriptor and ATM service category for the connection defines the traffic contract. These components and attributes define the desired transmit and receive characteristics, including the peak and sustained cell rate. The traffic contract is agreed upon between the service provider and the customer.

Traffic management strategies are based on the receive and transmit traffic descriptor parameters and the ATM service category. These strategies include

- connection admission control (CAC), which permits or rejects connection setup
- traffic shaping, which controls traffic in the transmit direction
- usage parameter control (UPC), which monitors and controls traffic in the receive direction
- buffer, queue, and congestion/discard management, which handle congestion situations as they arise

For more detailed information on these traffic management controls, see the following documents:

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*
- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

Connection configuration also includes fault management for VCCs and VPCs:

- VPT VPCs can be defined for fault management at the VP layer.
- The loopback setting determines if fault management initiates link-side and node-side periodic segment loopbacks as well as end-to-end loopbacks.
- The OAM segment boundary defines the connection as either a segment end point or a connecting point for connections defined as unswitched relay points.
- The fault hold off time determines whether VP-layer fault indications are passed to the VC layer.

For more information, see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

For Nortel Multiservice Switch 7400 nodes, performance monitoring can be activated, on a per connection basis, in order to calculate cell loss ratio (CLR) and availability ratio (AR). Performance monitoring can also be activated at the interface level, for each connection on the interface.These measurements

allow service providers to guarantee the quality of service provided to a customer. For more information on performance monitoring see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

Switched permanent virtual connections and paths

SPVCs and SPVPs differ from PVCs and PVPs in the following ways:

- Configuration for a connection is required at an end point only.
- Hop-by-hop configuration of a connection is not required. For nodes supporting UNI/IISP/AINI, you must configure network nodes for hop-by-hop routing. For nodes supporting PNNI, no additional configuration is required.
- Route selection and connection establishment is automatic.
- Re-establishment in case of a network failure is automatic.
- Support for the lock and unlock capability at the source end.
- Optional call redirection capability allows the specification of a secondary destination where the call can be routed in the event that establishment at the primary destination fails.

The following network features support SPVCs and SPVPs:

- PNNI signaling protocol
- UNI signaling protocol
- interim switch signaling protocol (IISP): SPVC and SPVP support is a Nortel Multiservice Switch node-specific value-added implementation.
- ATM inter-network interface (AINI)
- addressing
- ATM routing

SPVCs and SPVPs use a proprietary information element (IE) added to the call setup PDU sent across each IISP and AINI signaling channel. This IE specifies the destination VPI.VCI connection map address space for the SPVC.

The SPVC feature supports both VCCs and VPCs. Where required, a virtual path can be an unswitched connection.

See also Hitless ATM services for Multiservice Switch nodes (page 28) for further details.

AIS generation

Nortel Multiservice Switch nodes provide the option to enable OAM fault management (F4/F5 flow), on a per connection basis, for failed SPVC or SPVP connections through the continuous generation of an ATM OAM end-to-end AIS cell at the source and/or destination end. This functionality can be enabled or disabled independently at either the source or destination end of an SPVC or SPVP. If enabled, OAM AIS cells are generated at nominally one second intervals for the duration of the disconnect. In most cases, the OAM AIS cells would be received and processed by customer premise equipment (CPE).

Before the AIS cell generation capability can be initiated on the source SPVC or SPVP, the *aisGeneration* attribute on the source end must be enabled. For the destination end of the SPVC or SPVP, the destination end itself must be configured and the aisGeneration attribute must be enabled.

Modifying the *aisGeneration* attribute, whether to enable or disable AIS cell generation, is not service impacting. Configuring a destination SPVC or SPVP over an existing dynamic SPVC or SPVP results in the dynamic SPVC or SPVP becoming disconnected.

For details on how to configure a destination end, refer to NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

SPVC/P preemptive reestablishment

A race condition can occur within an ATM network during SPVC failure recovery. Depending on which link fails in the network, the terminating end may receive an SPVC reestablishment SETUP message before it receives the RELEASE message. If this occurs, the SETUP message will be rejected, resulting in the SPVC being released with a cause code of 34 (SPVC termination vpi.vci unavailable). Cause code 34 does not result in a RELEASE with crankback and the call goes onto a retry timer cycle (configured against the ATM interface). A comparison is now made between the calling party information in the SETUP message and the operational SPVC destination. If the SETUP is coming from exactly the same originator (NSAP address, vpi, vci), it is assumed that this is a race condition. The original call is released with a cause code of 16 and the new SETUP message is processed. This results in the SPVC remaining active.

For information on the cause codes, refer to NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

Switched virtual connections

ATM SVCs are virtual circuit data paths that run through the ATM network. However, unlike PVCs, SVCs are dynamically set up and torn down as required by subscriber applications. SVCs do not require configuration, but network nodes must be configured for SVC routing. Nortel Multiservice Switch nodes do not originate or terminate SVCs.

The following network features are provided to support SVCs:

- ATM addressing
- ATM signaling
- ATM routing

Most traffic management functions apply to both unswitched connections and switched connections including SPVCs, although some features do not apply at the VCC level.

For more information on traffic management, see the following documents:

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*
- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

See also Hitless ATM services for Multiservice Switch nodes (page 28) for further details.

Virtual path termination

Through configuration, you can define two types of virtual path terminations (VPT) as connection end points of a VPC:

- Basic VPT (page 26)
- Standard VPT (page 27)

Basic VPT

With a basic VPT, independent VCCs and VCCs within a basic VPT are equivalent from the perspective of the data path and traffic management. Basic VPT supports the following capabilities:

- VP operations and maintenance functionality (loopbacks and VP/VC fault interworking)

- aggregated VCC statistics and accounting
- optional VPT-CAC

All Nortel Multiservice Switch function processors support basic VPT.

Basic VPT provides one level of traffic management at the VC level only (that is, it does not provide VP-level shaping). For more information on traffic management, see the following documents:

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*
- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

Standard VPT

Standard VPT allows simultaneous traffic management at both the VP and VC levels. Standard VPTs can also dynamically share bandwidth among other connections on the interface. The VCCs within the VPT dynamically share bandwidth with the VPT. Nortel Multiservice Switch function processors with PQC and AQM ASICs support standard VPT.

In addition to the features supported by basic VPTs, standard VPTs also support other capabilities including VP shaping and weighted fair queuing (WFQ).

Unlike a basic VPT (in which there is no VP-related data path for ATM cells at the VPT connection point), a standard VTP is directly involved in the data path. This configuration allows most of the traffic management capabilities of relay point VPCs to be available at the VPT connection point.

Point-to-multipoint connections

In a point-to-multipoint connection, identical information is distributed unidirectionally from one calling end station to a set of selected called end stations. The calling end station is the root and the called end stations are the leaves.

Nortel Multiservice Switch 7400, Multiservice Switch 15000, and Multiservice Switch 20000 support four types of point-to-multipoint connections:

- point-to-multipoint PVCs
- point-to-multipoint PVPs

- point-to-multipoint SPVCs
- point-to-multipoint SVCs

See the table ATM connection types: support by function processor (page 17) for a summary of function processors supporting point-to-multipoint connections.

For information about configuring point-to-multipoint connections, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*. For more information on point-to-multipoint connections, see Point-to-multipoint connections (page 179).

Anycast point-to-point connections

The ATM anycast point-to-point capability allows an end station to connect to any member of a group at the far end using PNNI routing. The anycast capability is useful in network applications where an ATM address must represent a general service rather than a specific node or end station.

For more information on anycast capability, see Anycast point-to-point connections (page 188).

Hitless ATM services for Multiservice Switch nodes

For Nortel Multiservice Switch 15000 and Multiservice Switch 20000, the following ATM services offer a hitless service:

- PVCs (VCCs, VPCs and VPT VCC (basic and standard) connections)
- point-to-point SVCs
- source and destination SPVCs
- point-to-point SVPs
- source and destination SPVPs

A service is hitless when the software that provides the service can run uninterrupted, even when the hardware providing the service changes. This can occur during an equipment protection switchover, or a switchover during a software migration.

For the connections to be hitless, the ATM interface providing the service must be provisioned for equipment protection with either:

- one-for-one sparing for FPs with electrical interfaces
- dual-FP line APS for FPs with optical interfaces

When a service is spared, two instances of the service software are created to handle a single interface: the active instance and the standby instance.

The active instance creates protected connections on the active card. The standby instance creates identical protected connections on the standby card. The connections on the standby card do not handle traffic but are ready to immediately take over handling of both the ingress and egress traffic flows if the active card fails.

Operator commands for spared services are processed by the active instance only. However, provisioning data is processed in parallel by both the active and standby instances. Accounting records and statistics are generated and maintained for the active instance only.

For Multiservice Switch 15000 and Multiservice Switch 20000, optical FPs can run both spared and unspared ATM services. Both the active and standby card's connection resources are divided into a protected pool and an unprotected pool. Spared connections are allocated from the protected pool. Unspared connections are allocated from the unprotected pool. A network operator can provision the size of these pools. See NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management* for details.

See NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for a description of:

- hitless services
- hot, warm and cold standby applications and features
- one-for-one equipment sparing and two-LP sparing
- mixing spared and unspared services

Hitless services minimize the interruption of cell forwarding only. During an FP switchover, all applications that were running on the FP can lose administrative data even if that application is providing hitless services. This includes:

- performance statistics, such as cell counts
- partial accounting records and any accounting records that reside in the memory of the FP before the FP switchover
- unspooled accounting records from the last TODA change-over period
- the OSI state (A service that is locked before the FP switchover becomes unlocked after the switchover.)

Accounting data is initialized to zero if the standby instance becomes active.

If a connection is generating AIS cells prior to an FP switchover, it will be interrupted until near the end of the switchover, at which time it will resume AIS cell generation.

Attention: Virtual SPVCs do not support this functionality.

Conditions for hitless software migration for Multiservice Switch nodes

For a software migration to be hitless for ATM services, both ends of the connection across the Nortel Multiservice Switch 15000 and Multiservice Switch 20000 fabric must be spared. Such a connection is considered fully spared.

During the migration, fully spared connections remain operational with minimal cell loss. Unspared connections are lost. Partially spared connections, where only one end of the connection is spared, are also lost. Unspared and partially spared connections can only be re-established after the migration is complete.

See NN10600-270 *Nortel Multiservice Switch 7400/15000/20000 Software Installation* for a description of hitless software migration.

Behavior of Multiservice Switch node switched ATM services during an FP switchover

Switched ATM services, such as SVCs, dynamically produce routing and signaling data. When the switched ATM services are configured on Nortel Multiservice Switch 15000 and Multiservice Switch 20000 to provide hitless connections, this dynamic service data is automatically stored by both the active instance and the standby instance of the software. The active instance and the standby instance are synchronized to ensure that the standby instance can replace the active instance with minimal delay.

This section describes the behavior of switched ATM services, such as SVCs, during an FP switchover. The behavior of the ATM service can be broken down into the following topics:

- Physical and ATM layer switchover (page 30)
- Signaling interface and call processing switchover (page 31)
- ILMI channel switchover (page 31)
- Routing control channel switchover (page 31)

Physical and ATM layer switchover

For electrical FPs, the new hardware becomes active in less than 100 milliseconds. For optical FPs, the new hardware becomes active in less than 50 milliseconds.

In the egress direction, the ATM layer begins forwarding cells as soon as the new hardware becomes active. In the ingress direction, the other cards in the shelf must adjust to send cells to the new hardware. This occurs in less than 50 milliseconds, in parallel with physical layer switchover.

Signaling interface and call processing switchover

Because the physical and ATM layer switchover occurs within 50 to 100 milliseconds, the signaling interface to an adjacent node does not fail during a switchover.

However, synchronization of the signaling interface with the adjacent node is temporarily lost because the ATM interface on the new hardware restarts its signaling channel.

Synchronization is re-established without the adjacent node dropping any active calls provided the adjacent node behaves according to *User-Network Interface (UNI) Specification (3.0/3.1/4.0)* and *Private Network to Network Interface Specification (1.0)* from the ATM Forum technical committee.

The switchover does affect calls that are being established when the switchover occurs. This applies to calls being established by the peer signaling interface on the adjacent node, or calls being established by other FPs in the Nortel Multiservice Switch 15000 or Multiservice Switch 20000 node. Calls of this nature are cleared after the switchover.

ILMI channel switchover

Because the physical and ATM layer switchover occurs within 50 to 100 milliseconds, the ILMI channel does not fail during a switchover.

In addition, after switchover, the newly-active ATM interface software can respond to any ILMI messages immediately after the physical and ATM layers recover.

Routing control channel switchover

Because the physical and ATM layer switchover occurs within 50 to 100 milliseconds, the routing control channel (RCC) to an adjacent node does not fail during a switchover.

However, because PNNI topology and state routing information is stored on the CP instead of the FP, a switchover causes the newly-active ATM interface to effectively restart its RCC by re-staging with its neighbor. This causes the PTSE database and Hello data structures to be re-synchronized.

Synchronization is re-established without the adjacent node dropping any active calls provided the adjacent node behaves according to *Private Network-Network Interface Specification, Version 1.0* from the ATM Forum technical committee.

During the re-staging of the RCC with the neighbor, source PNNI nodes in the network do not route calls through the newly-active PNNI interfaces as a result of receiving the PTSE link down update.

The amount of time needed to perform neighbor re-staging for all PNNI interfaces depends on the total number of interfaces and the size of the PNNI network. Nortel Networks estimates that for 16 interfaces, all interfaces can re-stage the RCC within three seconds.

If a hitless software migration causes the switchover, the complete PTSE database must be exchanged between all peer nodes; not just re-synchronized. The amount of time required to do this depends on the size of the database.

Connection mapping

Connection mapping under the ATM interface depends on the function processor type. From the perspective of connection mapping, there are three types of function processor:

- ATM IP function processors (see ATM connection mapping for ATM IP function processors (page 32))
- 8-port CQC-based function processors (see Connection mapping for 8-port CQC-based function processors (page 33))
- APC-based function processors and 2- and 3-port CQC-based function processors (see Connection mapping for APC-based function processors and 2- and 3-port CQC-based function processors (page 33))

ATM connection mapping for ATM IP function processors

ATM IP function processors provide simplified connection mapping features that conform to standards-based approaches.

ATM IP function processors have the following connection map features:

- Connection mapping does not require extensive configuration.
- ATM IP function processors support 12 bits for VPIs, yielding a maximum of 4095 VPIs in the connection map space. These function processors support intelligent defaults for the maximum number of VPI bits. Service providers can override defaults as required.
- The entire VPI.VCI space is available for connections, limited only by the number of bits configured for VPIs. VCIs can range from 32 to 65 535. If

the service provider uses 12 bits, VPIs can range from 0 to 4095. If the service provider uses 8 bits, VPIs can range from 0 to 255.

- There is no separation of the VPI.VCI space into regions reserved for VPCs and regions reserved for VCCs.
- Configuration of regions of the VPI.VCI space for allocation of switched connections conforms to the *Integrated Local Management Interface (ILMI) Specification Version 4.0*:
  - SVPCs can use VPIs from 1 to a maximum configured value.
  - SVC connections can use VPIs from 0 to a maximum configured value, and VCIs from a minimum configured value up to 65535.

Connection mapping for 8-port CQC-based function processors

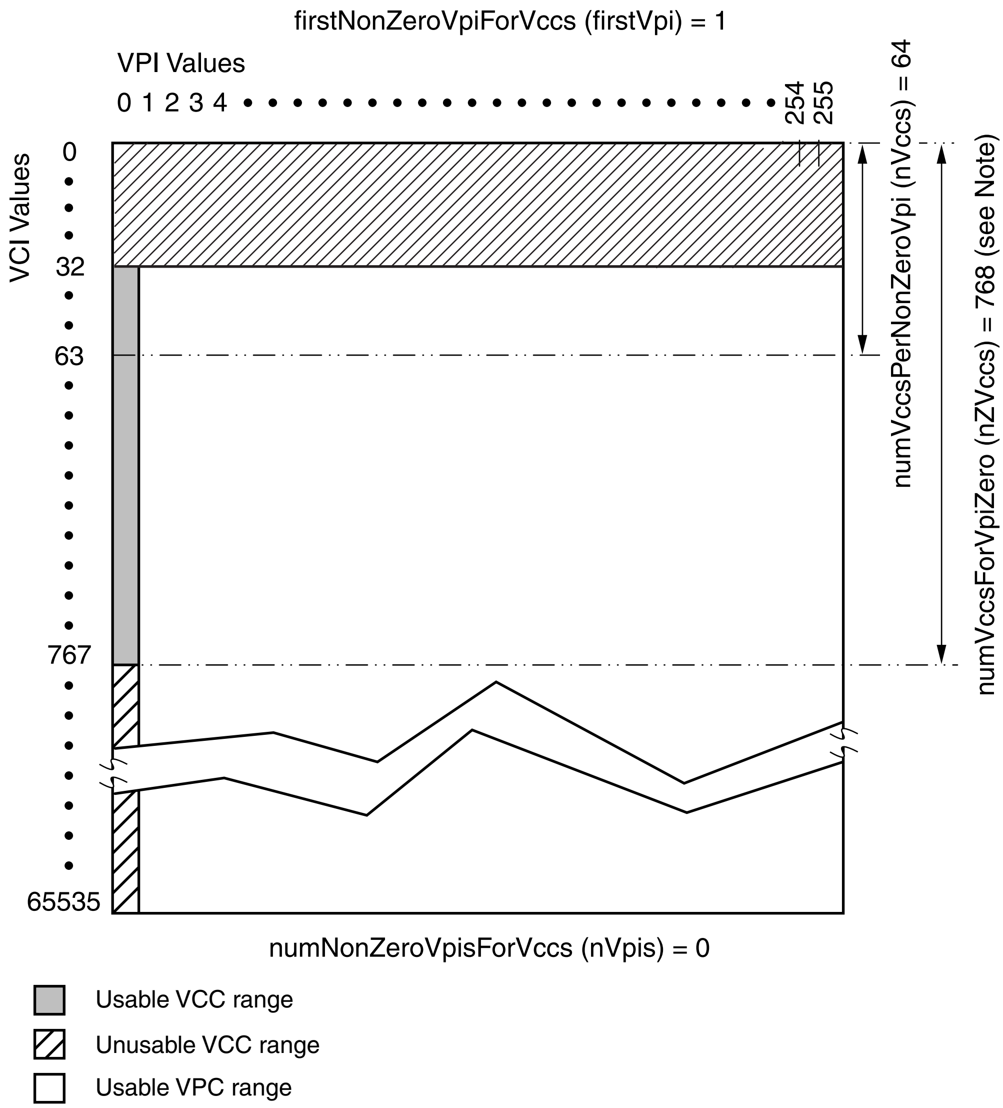
Eight-port CQC-based function processors have the same characteristics as ATM IP function processors with the exception that they do not support VPI number 4095.

Connection mapping for APC-based function processors and 2- and 3-port CQC-based function processors

A properly configured connection map is essential for proper operation of ATM interfaces in a Nortel Multiservice Switch ATM network. The recommended approach is to use the same connection mapping configuration for every ATM interface in a Multiservice Switch network. This approach both simplifies configuration and ensures that connection instance values are not mismatched on opposing ends of an ATM interface.

The model used to define connection mapping for an ATM interface is shown in the figure ATM interface connection map for 2- and 3-port CQC-based FPs: default values (page 34). For a complete technical description on connection mapping, see Connection mapping (page 204). For configuration information, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

ATM interface connection map for 2- and 3-port CQC-based FPs: default values



Notes: The variable nZVccs is equal to 768 because its range is 0 to 767.
The VCI range of 0-31 is reserved for use by future standards.

PPT 3094 001 AA

ATM network addressing

In an ATM network, addresses uniquely identify source and destination points. Use the following sections to learn more about ATM network addressing.

Navigation

General considerations for ATM addressing

The ATM Forum defines the standards and requirements for addressing in an ATM network. For complete information, see *ATM Forum Addressing: Reference Guide*, STR-RA-ADDR-01.09.

How a service provider handles addressing in a Nortel Multiservice Switch network depends on the type and mix of connections on each interface. Possible connections on an interface include the following:

- permanent virtual connections (PVC)
- switched permanent virtual connections (SPVC)
- switched virtual connections (SVC)
- permanent virtual path (PVP)
- switched permanent virtual path (SPVP)
- switched virtual path (SVP)

The connection map address space is the basis for all addressing in a Multiservice Switch network. The connection map is the range of virtual path indicator (VPI) values and virtual channel indicator (VCI) values that are available on an interface to terminate connections. These values define the VPI and VPI.VCI addresses in the connection map. How a node and the network use the connection map address space depends on the type of function processor, the type of connection, and the current traffic demands on the interface.

For more information on considerations for ATM addressing, see the following sections:

- General addressing characteristics (page 36)
- Considerations for ATM IP function processors (page 37)
- Considerations for CQC-based function processors (page 37)

General addressing characteristics

Nortel Multiservice Switch nodes support five interface types:

- basic interface
- user-to-network interface (UNI)
- interim inter-switch signaling protocol (IISP) interface
- ATM inter-network interface (AINI)
- private network-to-network interface (PNNI)
- virtual interfaces (through PVPs)

A connection is an PVC, SPVC, SVC, PVP, or SPVP. For PVCs and PVPs, you define a specific address in the connection map (all four interface types support PVCs and PVPs). The connection remains in service and available as long as the links and interfaces remain in service and you do not reconfigure the connection.

UNIs, IISPs, AINIs and PNNIs support network service access point (NSAP) addresses. For AINIs and IISPs, you configure addresses (static addresses). For UNIs, you configure static addresses, enable address registration through integrated local management interface (ILMI), or both. For PNNIs, you configure summary addressing to support the dynamic routing protocol. PNNIs need static address configuration only for SPVC and SPVP termination points.

On call setup for SPVCs, SVCs, SPVPs, and SVPs, the nodes in the connection assign and accept the VPI.VCI address (for SPVCs and SVCs) or the VPI address (for SPVPs and SVPs) in the connection map address space, according to the signaling and routing protocol rules that govern the interface.

Each UNI, IISP, AINI, and PNNI interface has a default address. Further, you can also configure one or more additional addresses for an interface. An ATM device associated with an additional address can be located immediately on the other side of the interface. The device can also be located several hops along the connection. In this way, the network records that one or more additional addresses are somehow reachable through the interface.

Considerations for ATM IP function processors

The connection map for ATM IP function processors does not require configuration for the address space. The entire connection map is available. Connections over these function processors select the correct address in the connection map space. For more information on connection mapping for ATM IP function processors, see ATM connection mapping for ATM IP function processors (page 32).

Considerations for CQC-based function processors

For each interface on a CQC-based function processor in the network, you configure the usable range of addresses in the connection map. This usable range defines the available VPI.VCI addresses for the interface. For example, if a node has five ATM interfaces, you configure five connection map address spaces. In this example, each connection map can be completely different, or two or more can be the same or very similar. For more information on the connection map, see Connection map address assignment for CQC function processors (page 58) and Connection mapping (page 204).

NSAP address formats

The NSAP address format consists of three main parts:

- a 26 hexadecimal digit prefix, known as the network prefix
- a 12 hexadecimal digit identifier, known as the end system identifier (ESI)
- a 2 hexadecimal digit selector (SEL)

The figure NSAP address format, showing network prefix and end system identifier (page 38) shows the structure of the NSAP address.

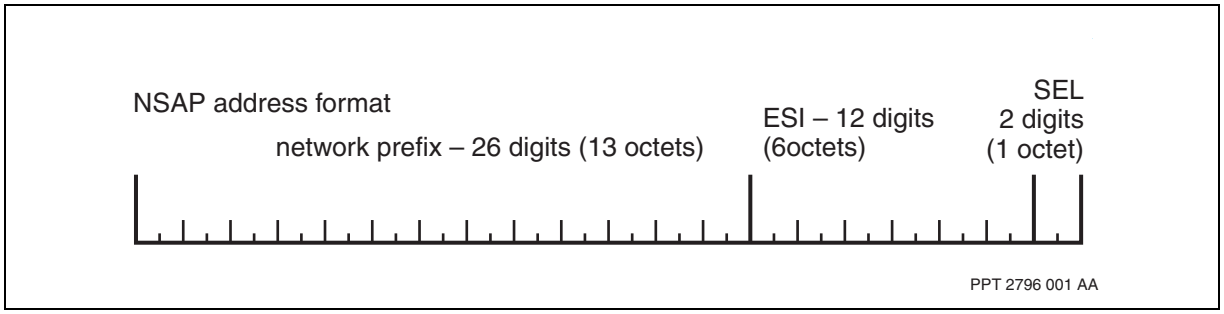
Nortel Multiservice Switch nodes support addresses and prefixes for the following ATM Forum NSAP address formats:

- data country code (DCC)
- international code designator (ICD)
- E.164

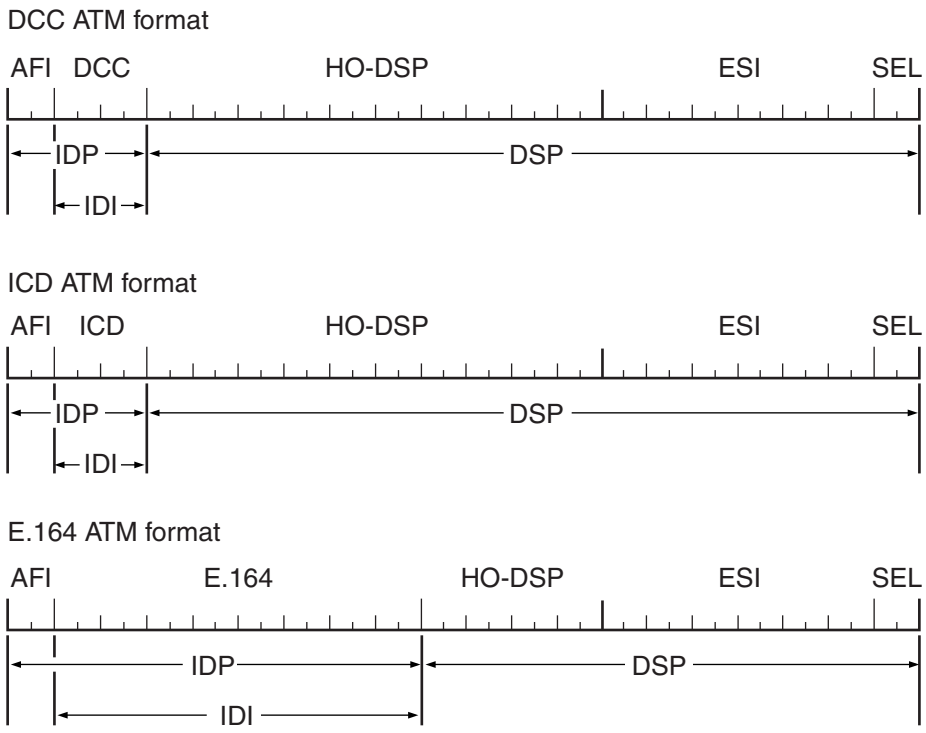
Nortel Multiservice Switch devices also support native E.164 addresses. See Native E.164 addresses (page 42).

The figure NSAP address formats, by address type (page 39) shows the detailed structure of each format.

NSAP address format, showing network prefix and end system identifier



NSAP address formats, by address type



Legend

AFI = authority and format identifier
DCC = data country code
DSP = domain specific part
ESI = end system identifier
HO-DSP = high-order domain specific part
ICD = international code designator
ICI = initial domain identifier
IDP = initial domain part
SEL = selector

PPT 2449 001 AA

For more information on address formats, see the *User-to-Network Interface Specification Version 3.0*.

UNI, IISP, AINI and PNNI also support X.121-based NSAP addresses for frame relay. You can configure X.121-based NSAP static addresses under all three interfaces. You cannot configure X.121 prefixes because the ILMI protocol does not support the X.121 authority and format identifier (AFI).

Data country code format

The DCC format specifies the country in which an address is registered. The field length is two octets. DCCs are encoded in binary coded decimal (BCD) syntax. The codes are left-justified and padded on the right with the hexadecimal value F to fill the two octets.

International code designator format

The ICD format identifies an international organization (the British Standards Institute maintains the registration authority for ICD). The field length is two octets. ICDs are also encoded in BCD syntax. The codes are left-justified and padded on the right with the hexadecimal value F to fill the two octets.

E.164 encapsulated format

The E.164 format specifies integrated services digital network (ISDN) numbers. These numbers include telephone numbers. Nortel Multiservice Switch devices implement the international format of these numbers. Numbers are up to 15 digits long. The field length is eight octets. E.164 address formats are also encoded in BCD syntax. The address specification defines a leading semi-octet 0000 in the address to obtain the maximum length (15 digits). A semi-octet value 1111 also pads the address after the final semi-octet to obtain an integral number of octets.

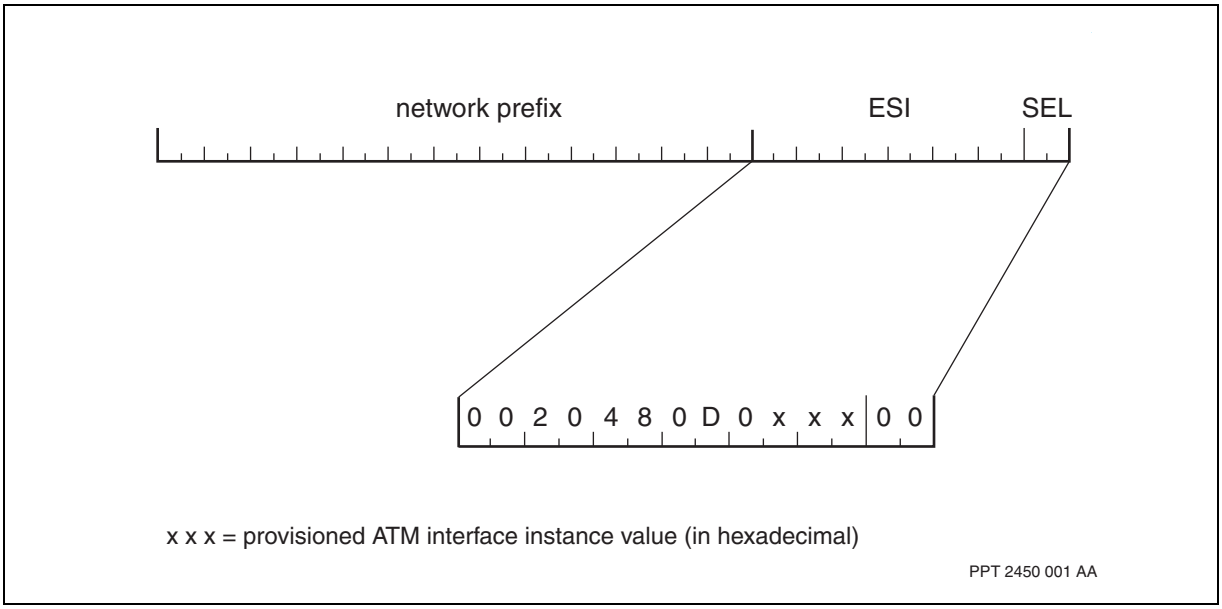
Multiservice Switch implementation of NSAP addresses

The Nortel Multiservice Switch default address follows the NSAP format as follows:

- The network prefix is the value for the node address prefix that you configure in the *nodePrefix* attribute under the *ModuleData* component. All interfaces on a Multiservice Switch device use the node address prefix as the network prefix in the default address.
- The ESI is a 12-digit value that the system assigns. Note that the ESI incorporates the configured value of the ATM interface. The figure Default address format (page 41) illustrates the format of the ESI.
- The selector is a 00 hex value in the Multiservice Switch node implementation.

Typically, you use the default address as the destination address of an SPVC. You can also use a customer-defined static address (either partial or complete) for the destination. You configure customer-defined static address at the destination node. You cannot use registered addresses to set up SPVCs.

Default address format



For more information on the Multiservice Switch device implementation of NSAP addresses, see the following sections:

- NSAP addresses and node configuration (page 41)
- Native E.164 addresses (page 42)
- NSAP to native E.164 conversion (page 43)
- Native E.164 to NSAP conversion (page 43)
- Sub-addresses (page 43)

NSAP addresses and node configuration

When you configure an NSAP address, you can enter the complete 40-digit address in NSAP dotted notation. The following is the general format for an NSAP address:

```
AFI.IDI.HO-DSP.ESI.SEL
```

The authority and format identifier (AFI) is a two-digit code that identifies the authority allocating the initial domain identifier (IDI). The AFI also indicates the format for the remainder of the NSAP address. Nortel Multiservice Switch nodes recognize the following AFI codes:

- 37, which indicates an X.121 format
- 39, which indicates a DCC format
- 47, which indicates an ICD format
- 45, which indicates an E.164 format

The IDI is between 4 and 16 hexadecimal digits long, depending on the address format, as defined in the section NSAP address formats (page 37).

The high order domain specific part (HO-DSP) ranges from 8 to 20 hexadecimal digits long. The AFI, IDI, and HO-DSP portions of the address make up the network prefix.

The last two portions of the NSAP address identify the end system. The end system identifier (ESI) is 12 digits long and uniquely identifies the end station. The ESI can be a globally unique identifier such as a media access control (MAC) address. The selector (SEL) is a two-digit code that end systems use.

Here is an example of a 40-digit E.164 NSAP address:

```
4500006137222222F000000000020480D001100
```

For more information on the NSAP address data type, see NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference*.

Native E.164 addresses

Although Nortel Multiservice Switch nodes use NSAP addresses within the network, nodes can also handle an incoming call setup protocol data unit (PDU) that uses a native E.164 address as a destination address.

The native E.164 address is NSAP-encapsulated before processing. The switch restores the address field to its native E.164 format as the call setup PDU exits the Multiservice Switch node on the transmit link. This process is the default mode of operation, which preserves the format of the E.164 address in the call setup PDU.

Also through configuration, you can specify that all outgoing call setup PDUs on a particular UNI, IISP, or AINI interface must have an NSAP destination address or a native E.164 destination address. If this specification exists, the node converts the address before the PDU passes to the transmit link.

When the destination address must be an NSAP address, a PDU with a native E.164 address converts to an NSAP-encapsulated address. This conversion is necessary, for example, on a connection to a device that does not handle native E.164 addresses.

When the destination address must be a native E.164 address, a PDU with an NSAP address converts to a native E.164 address, if possible. Only NSAP-encoded E.164 addresses can convert to the native E.164 format. This conversion is necessary, for example, on a connection to a device that handles only native E.164 addresses (such as a public-network switch). If the switch applies E.164 address conversion, a call setup request fails if there is no possible way to convert the address to native E.164.

For more information on the two types of conversion, see the following sections:

- NSAP to native E.164 conversion (page 43)
- Native E.164 to NSAP conversion (page 43)

NSAP to native E.164 conversion

The switch copies the called party number NSAP E.164 address into the called party sub-address IE. The copy overwrites any sub-address information that existed previously in the IE. The called party number address becomes the E.164 portion of the NSAP address without the domain specific part (DSP). If the called party number is either ICD or DCC NSAP format, the system clears the call.

Native E.164 to NSAP conversion

If the sub-address IE contains an NSAP address, the NSAP address from the sub-address IE overwrites the called party number address. This process clears the sub-address. If there is no sub-address IE, the native E.164 address converts to NSAP format with zeros in the DSP portion.

If a conversion cannot take place, the system returns the Invalid number format error (UNI 3.0/3.1 cause code 28). Conversion cannot take place on an address in ICD or DCC NSAP format, or on address where you do not configure conversion for the interface.

Sub-addresses

Nodes access sub-address IEs and carry them transparently across the network, except in the address conversion scenarios.

NSAP information query

You can view NSAP information by provisioning unique node and interface names. These names enable you to monitor and troubleshoot PNNI network nodes. The node and interface names are displayed as a combination of the PNNI node name, ATM interface name, and logical link. The NSAP information of each node can be obtained by querying the *nodeName* attribute.

For information about:

- provisioning node and interface names, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*
- displaying NSAP information, see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*

Address registration through ILMI

At the UNI, the ILMI specification provides

- status, configuration, and control information about the link and the physical layer parameters at the UNI
- dynamic address registration across the UNI

You configure ILMI capabilities on UNIs only. IISPs, AINIs, PNNIs, and basic interfaces do not have ILMI capabilities.

The ILMI communication protocol is SNMP/AAL5. The ATM UNI management information is in a management information base (MIB). The types of ATM UNI MIBs are

- physical layer
- ATM layer
- ATM layer statistics
- virtual path layer
- virtual channel layer
- address registration information

Requests for management information other than dynamic address registration as defined in the ATM-FORUM-ADDR-REG MIB triggers a noSuchName response. The system sends this response to the device that originated the request. Section 5.8.6 of the *User-to-Network Interface Specification Version 3.1* defines the ATM-FORUM-ADDR-REG MIB.

Through configuration, you can disable ILMI address registration capabilities on any specific interface. You can use this approach when the network administrator does not want users to register their own addresses. Note that Nortel Multiservice Switch node does not initiate queries for examination and synchronization of remote addresses.

For more information on the ILMI MIB specification, please see section 4 and section 5.8 of the *User-to-Network Interface Specification Version 3.1*.

For more information on the Multiservice Switch address registration capabilities, see the following sections:

- ILMI address registration capabilities (page 44)
- ILMI configured options (page 46)

ILMI address registration capabilities

The table Summary of ILMI address registration capabilities (page 45) describes the ILMI address registration capabilities at the UNI.

Summary of ILMI address registration capabilities

Capability	Description
ColdStart Trap sent after start-up or restart	The ColdStart Trap PDU alerts the receiver that the remote side has started up or restarted. Upon receipt of the ColdStart Trap PDU, the receiving node re initializes its address table to empty.
Initialization-time exchange of address information	These procedures allow the user and the network to exchange addressing information. The protocol registers the ATM addresses at the UNI and the address then becomes part of the signaling messages. This exchange includes the capability of the network side to support more than one network prefix.
Dynamic addition/deletion of addresses	The user-side of the UNI can add and delete addresses and can inform the network of these address changes through the ILMI address registration MIB. The network side can add and delete network prefixes and communicate these changes to the user-side through the ILMI setRequest PDU. If the user-side receives a delete prefix request, the user-side immediately de-registers all addresses that it registered by combining that prefix with its ESIs. Whenever the interface receives a response that is not NOERROR, it restarts the ILMI channel.
Address de-registration on the UNI on a port down event	When the UNI is down, the system de-registers addresses for that UNI.
Periodic polling	The system issues periodic polls to detect loss of ILMI connectivity, and issues an ILMI request message every 5 seconds. When it receives no ILMI response messages for 4 consecutive polls, the system declares the UNI down.
Encoding and decoding of ILMI address registration messages	The system encodes/decodes ILMI PDUs as SNMP (Version 1) setRequest messages. The ILMI PDUs refer to address objects cited in the ATM-FORUM-ADDR-REG MIB. Any other ILMI PDU triggers a noSuchName response.
One VCC for sending AAL5-encapsulated ILMI SNMP messages	The node uses this VCC for requests, responses, and traps. The default VCI value for the VCC is VCI=16, but you can reconfigure this default. The VCC always uses VPI=0.
Retransmission of SetRequest messages	SetRequest messages are retransmitted if the node does not receive a Response message within the time-out period. If the number of retransmissions reaches a defined threshold, the link is down.

(1 of 2)

Summary of ILMI address registration capabilities (continued)

Capability	Description
ILMI address registration for ATM Forum UNI 3.0, ATM Forum UNI 3.1 and ATM Forum UNI 4.0 is supported	UNI Versions 3.0, 3.1, and 4.0 are compatible for the ILMI address registration protocol.
UNI group addressing supported for UNI 4.0 and PNNI 1.0	UNI version 4.0 and PNNI version 1.0 are compatible with ILMI 4.0 group addressing capabilities.
(2 of 2)	

ILMI configured options

The three configured options for the ILMI operating mode are:

- ILMI disabled—This option provides no ILMI capabilities for the UNI. The ILMI channel is unconnected.
- ILMI dynamic address registration disabled—This option provides link monitoring capabilities only. The ILMI staging between the user side and network side takes place. If the UNI is network side, the switch sends its node prefix to the user side.

  If the user side tries to register any addresses, the network side accepts the registration but does not put the address in the routing tables. If the UNI is user side, the user side does not care if the network registers a prefix within the 5 second time frame as specified in the UNI specification. The UNI accepts any registered prefixes but does not initiate address registration. Periodic polling monitors the channel for both network and user side configurations.
- ILMI dynamic address registration enabled—This option provides full address registration capabilities as outlined in this section.

Static addressing for UNI, IISP, and AINI

You can manually assign one or more addresses that end systems can reach through a particular UNI, IISP, or AINI interface. You configure these addresses on an interface basis.

If the UNI does not support ILMI address registration, or when the UNI needs additional exception addresses, use static addresses. For IISP and AINI interfaces, use static addresses to assign static routes.

A configured static address does not need the entire 40 hexadecimal digit NSAP address, and can have less than 40 digits. When address length is less than 40 digits, the address refers to all addresses that start with the specified digits. That is, one static address entry can encompass a range of addresses. For example, if 391111 is a static address on an IISP interface, all addresses that start with 391111 are reachable through that interface. This process is known as best-match addressing.

You can configure static addresses as either primary or alternate addresses. A primary address indicates that the interface belongs in the list that call setup uses first when selecting a route to the address. An alternate address indicates that the interface belongs in the list that call setup uses only when the primary list provides no results.

Nortel Multiservice Switch ATM networks also support wild card addresses. A wild card address has a single ? character and represents all addresses (networking can reach all addresses through the associated interface). If a node can set up a call across several interfaces, the call setup process prefers those interfaces with specific addresses over interfaces with a wild card address.

In addition to default destination addresses, customer-defined static addresses can terminate an SPVC or an SPVP call setup request on UNIs, IISPs, AINIs and PNNIs. This application of a static address is a terminating address. Addressing works in the same way as SPVCs and SPVPs that use default addresses. The difference is that the terminating address is configured on the destination node. As a result, any SPVC or SPVP call setup request terminates the SPVC or SPVP call setup at that port. This result assumes that the destination address is a best match to the customer-defined static address. This process applies to static addresses of all lengths.

Address screening for UNI, IISP, and AINI

Address screening is a mechanism that enables Nortel Multiservice Switch devices to validate point-to-point and point-to-multipoint ATM signalling connections (SVCs, SVPs, SPVCs, and SPVPs) at ingress signaling interfaces (UNI, IISP, or AINI) based on the source and/or destination address of an ATM call connection request. You can configure specific addresses to be accepted or rejected for each UNI, IISP, or AINI interface.

Address screening allows you to provide access security for subscribers who use SVCs over a shared ATM network. It also allows you to block subscriber access to your internal network elements.

When calling address screening is enabled, incoming calls are accepted or rejected based on the call's source address. Likewise, incoming calls are accepted or rejected based on the call's destination address when called address screening is enabled. The Multiservice Switch node records the number of calls that are accepted, rejected, and unmatched.

Group addressing

Nortel Multiservice Switch nodes support group addressing for anycast point-to-point connections. The format of an ATM group address is the same as the format for an individual ATM endpoint address. Address registration relies on ILMI 4.0.

For more information on group addressing in the context of anycast point-to-point connections, see the following sections:

- ATM group address format (page 189)
- Characteristics and scope of group membership (page 190)
- ATM group addresses and PNNI (page 191)
- Group address registration (page 192)

PNNI addressing

The topology of a PNNI network is hierarchical. The peer group identifiers that the protocol uses for the routing domain define the structure of the hierarchy. The structure of the hierarchy, in turn, determines how you assign addresses to members of the peer group.

For more information on PNNI addressing, see the following sections:

- Peer groups (page 48)
- Node addresses (page 51)
- End system addresses (page 52)
- Address summarization (page 52)
- Address advertisement scope (page 54)

Peer groups

A peer group is a set of logical nodes that the network engineer groups together for the purposes of creating a routing hierarchy. A Nortel Multiservice Switch node can assume any role in a peer group:

- act as a border node
- act as a peer group leader at any level in the hierarchy
- represent the peer group as a logical group node (LGN) at the higher level
- act as a peer to the LGN at the lowest level

For more information on peer groups and hierarchy, see the ATM Forum's *Private Network-Network Interface (PNNI) Specification Version 1.0*.

When you assign addresses to nodes in a PNNI peer group, you create an addressing scheme that corresponds to the topological hierarchy. The figure Example of a PNNI peer group hierarchy (page 50) shows a simple hierarchy with three levels. In the Multiservice Switch scheme, the highest level is level 0 and the lowest level is level 104.

An address prefix that you assign to a node indicates that all addresses that begin with that prefix are reachable through that node. This addressing approach permits address summarization and therefore effective and efficient

scaling. As you move up the hierarchy, the prefix becomes shorter to encompass larger numbers of nodes within the logical groups. The figure Example of addressing for a PNNI peer group hierarchy (page 51) shows an example of addressing within a peer group hierarchy.

The example in the figure Example of addressing for a PNNI peer group hierarchy (page 51) shows how logical node A.1 is a LGN, and represents (and is configured on) the actual node A.1.2. which is a PGL in the peer group A.1. Logical node A.1 advertises at the higher level on behalf of the PG(A.1) peer group. Node A.1.2 is also a border node for the peer group, through which all other nodes in the PG(A.1) peer group are accessible to the network. The logical node and the border node functions are on a single node in this example, but can be shared by two separate nodes.

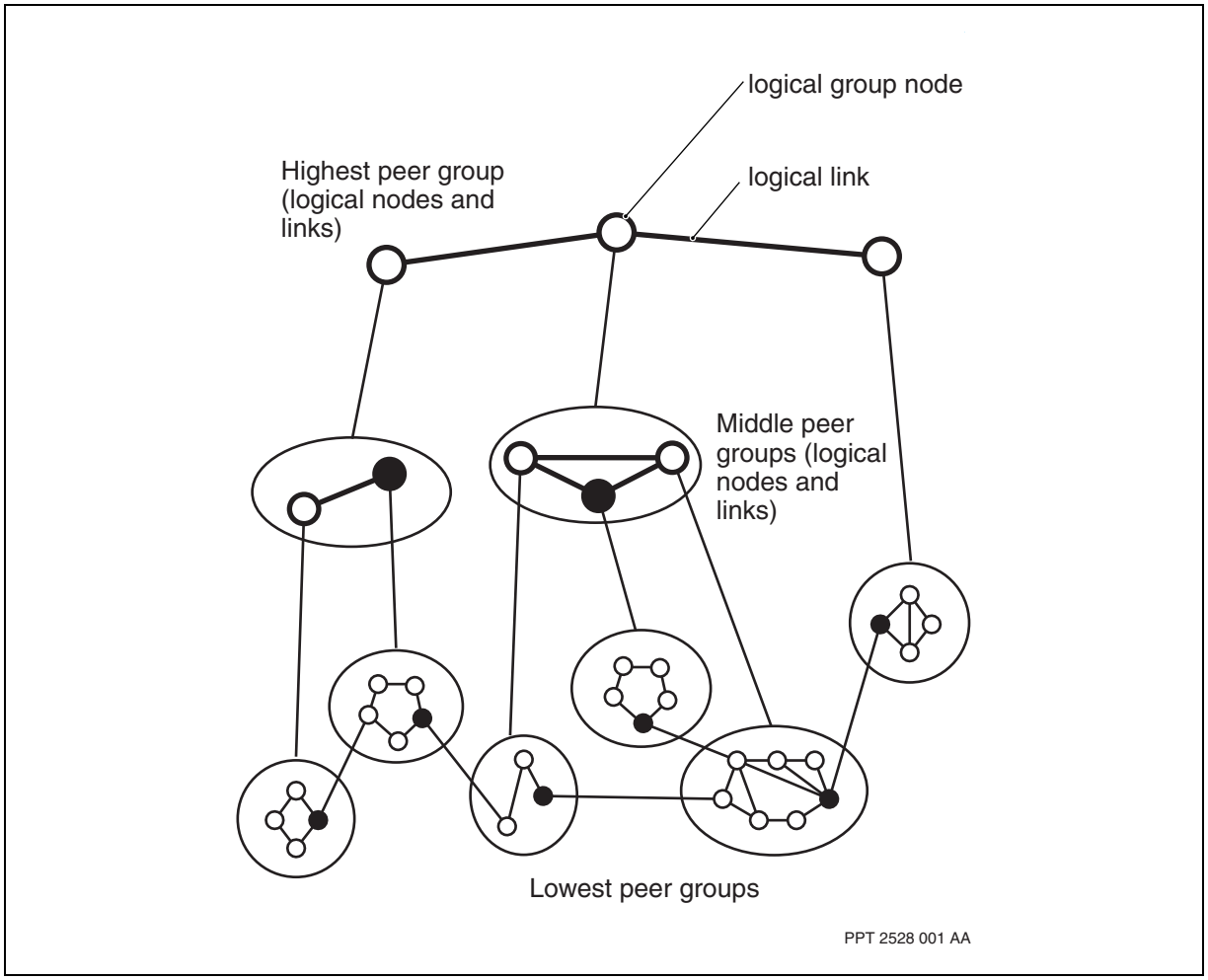
Logical node A.2 is also a PGL at peer group A, and represents (and is configured on) the actual node A.2.3, where A.2.3 is

- the border node through which all other nodes with prefix A.2 are accessible
- the PGL at peer group A.2

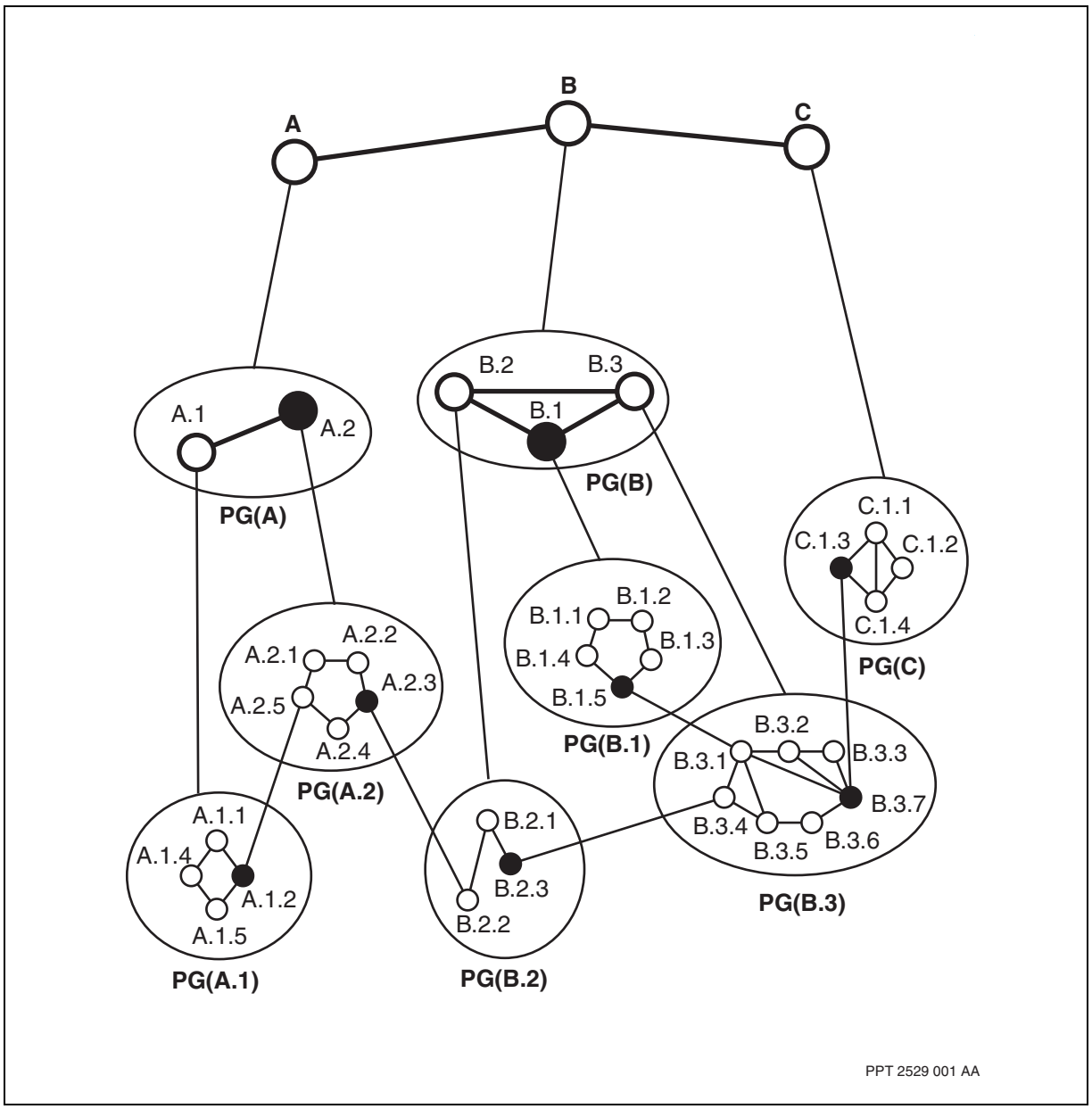
For any node, you can also include addresses that are exceptions in the hierarchy. Identify these exceptions with longer address prefixes.

PNNI routing uses only the first 19 octets of the end system identifier. The selector octet (octet 20) identifies destinations that are reachable through the same interface.

Example of a PNNI peer group hierarchy



Example of addressing for a PNNI peer group hierarchy



Node addresses

Use ATM end system addresses to identify nodes that participate actively in PNNI routing. The default static address for a node is based on the node ID and the MAC address. You assign a unique ATM end system address, known as the *nodePrefix* attribute, to each node. The PNNI protocol uses this unique address to establish switched virtual channel connections (SVCs). This unique address is created when a cfg/x level is provisioned.

End system addresses

PNNI advertises reachable end system addresses by using the prefixes of end system addresses to form address summaries. If an end system attached to a node does not fit into one of the node's configured summaries (foreign address), an explicit advertisement including all 19 octets (152-bit) will be displayed.

True summaries of reachable end system addresses have prefix lengths of less than 19 octets. Where the addressing hierarchy follows the topological hierarchy, you can use a single prefix to advertise reachable addresses to a large number of end systems. Shorter prefixes (that is, lower prefix length values) summarize greater numbers of addresses.

Routing calculations can include end systems in networks that support E.164 addresses only. To accommodate these configurations, Nortel Multiservice Switch nodes support advertising prefixes of E.164 addresses using the embedded E.164 NSAP format.

Address summarization

Address summarization is a feature of PNNI networking. Address summarization reduces the amount of addressing information that nodes advertise across the network.

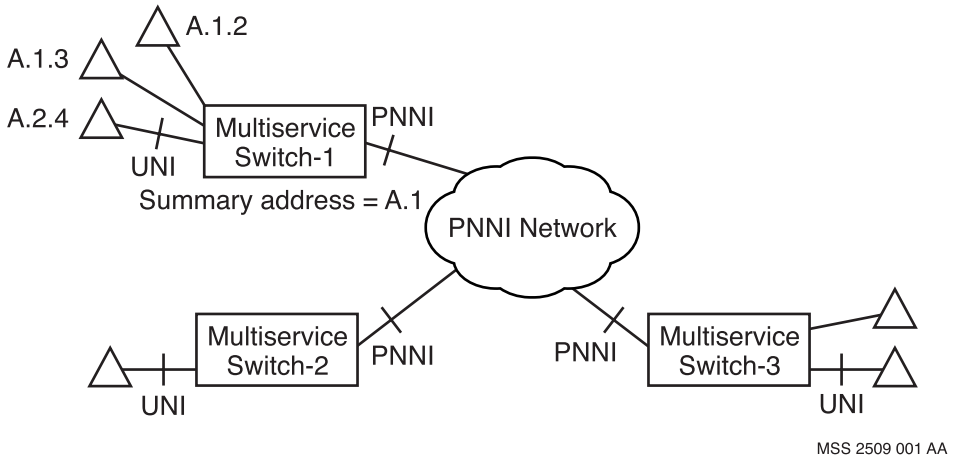
Summaries at higher levels can enclose other summaries at lower levels, so many summaries can match to varying degrees a given destination address. PNNI route computation directs calls to a logical node that advertises the best match for the given destination and that has a wide enough scope for the call. The best match is the matching (summary) advertisement with the longest prefix.

To implement summarization on a PNNI node, you define one or more address prefixes for the Nortel Multiservice Switch node. Each prefix represents a set of addresses that begin with that prefix and are available through that node. A prefix can match either end systems or nodes. A reachable address prefix is

- a summary address, which is an address prefix that you explicitly configure at that node or that assumes a default value
- a foreign address, which is an address that does not match any of the node's summary addresses

In contrast to these address types, a native address matches one of the node's summary addresses (see Native E.164 addresses (page 42)). The figure Example of address summarization (page 53) shows an example of address summarization.

Example of address summarization



In this example, the service provider configures Multiservice Switch-1 with the address prefix A.1 as a summary address. The interface advertises this prefix to the rest of the network. The advertisement indicates that any end system or node with an address beginning with A.1 is available through Multiservice Switch-1. In this example, Multiservice Switch-2 and Multiservice Switch-3 update and process only one address instead of two addresses (A.1.2 and A.1.3). The efficiency of this approach to addressing becomes more apparent as the number of represented nodes increases. Further, the network cannot include an end system address of A.1.1 that is accessible through Multiservice Switch-3. An end system with address A.1.1 must be accessible through Multiservice Switch-1.

The PNNI protocol uses a level indicator to determine the scope of the summary advertisement. This indicator corresponds to the PNNI routing hierarchy. The result is that the network advertises the summary address to all nodes with a level that is lower than or equal to the level indicator. See Address advertisement scope (page 54).

When you configure a summary address on a Multiservice Switch node, zero pad the address entry. For example, the hexadecimal summary address 471234 requires the following zero padding, which includes 32 zeros to bring the address entry to the required length.

```
47123400000000000000000000000000000000
```

When you configure the summary address on a Multiservice Switch node, define the prefix length (defined as the number of bits) for the summary address. In the above example, the prefix length is 24 bits (6 hexadecimal values times 4 bits for each value). If you define the scope for the summary

address as 20 (5 hexadecimal values times 4), the entry fails the check prov process because the Multiservice Switch software expects a 0 as the sixth hexadecimal value.

For more information on LGN addressing, see Logical group node representation (page 103)

Address advertisement scope

You can associate the advertisement scope with addresses and address prefixes in PNNI networking. The address scope, also referred to as the advertisement scope, defines the level of advertisement for an address, where the level of scope corresponds to a level of a peer group in the PNNI routing hierarchy.

The level indicator specifies the address scope of reachable addresses. The indicator specifies that the interface advertises addresses up to this level, but not to a higher level of PNNI routing. The highest level is 0, and the lowest level is 104. If the level is zero, then the address scope is unlimited and the node advertises the address throughout the PNNI routing domain.

There is a special case in which scope can be a value of -1. The Nortel Multiservice Switch node automatically derives this value as an operational attribute, based on the summary addresses and the addresses configured on the node. A scope of -1 indicates that the scope of a summary address is unknown. This case occurs when the summary address does not summarize any of the addresses currently configured on the node. When the scope is -1, the node sets the summary address state to inactive.

Addresses are eligible for advertisement or summarization into a peer group only if they have a address scope that is higher than or equal to that of the peer group. If there are addresses in the summary with different address scopes, the summary address takes the scope of the unsuppressed address with the highest scope.

Address suppression prevents the advertisement of addresses that match this prefix, regardless of scope. Where the scope of an address advertisement is wide enough to be advertised at a higher level (that is, the value of the address scope is smaller than the value for the level of the node), address suppression prevents it advertisement. A node can have an arbitrary number of suppressed summary addresses.

The scope of a PNNI advertisement for an address that is registered with an organizational scope at the UNI is a network-specific mapping. Rules for address suppression take precedence over rules for scope.

For more information on LGN addressing, see Logical group node representation (page 103)

Addressing considerations for point-to-multipoint connections

The addressing requirements for point-to-multipoint SPVC and SVC connections are identical to those for point-to-point SVCs. There are no additional considerations.

For more information on point-to-multipoint SPVC and point-to-multipoint SVC connections, see Point-to-multipoint connections (page 179).

Addressing considerations for virtual interfaces

Virtual UNI, IISP, AINI, and PNNI interfaces support all ATM end system address formats, including encapsulated NSAP X.121. Service providers can configure static addresses for each virtual UNI, IISP interface, AINI, and PNNI, including primary and alternate routing selection criteria. Dynamic address registration is not available, since ILMI is not supported for virtual UNIs.

Actual UNIs, IISPs, AINIs, and PNNIs have a default address that you can use to terminate SPVCs. Nortel Multiservice Switch software generates this default address when you configure an actual interface. In contrast, Multiservice Switch software does not generate a unique default address for a virtual interface. To configure a virtual interface to terminate an SPVC, configure a static address and prefix with a *TerminateSPvpAndSPvc* component.

If an SPVC on a virtual interface has no configured calling party address, the node omits the calling party information element (IE) from the setup PDU. This behavior is different from that on an actual interface which, in this scenario, uses the default address as the calling address.

ATM signaling

Nortel Multiservice Switch devices use standards-based signaling for connections over user-to-network interface (UNI), interim inter-switch protocol (IISP) interface, ATM inter-network interface (AINI), and private network-to-network interface (PNNI).

Navigation

Overview of signaling in a Multiservice Switch network

Signaling refers to the procedures that Nortel Multiservice Switch nodes follow to dynamically establish, maintain, and clear connections across UNI, IISP, AINI, and PNNI interfaces. Basic ATM interfaces do not support dynamic connections, and by extension they do not need to support routing and signaling.

The discussion in this section refers to UNI, IISP, AINI, and PNNI interfaces only, unless the material includes specific reference to basic interfaces.

Each interface has a signaling VCC, over which nodes send and receive signaling protocol data units (PDU) across the interface. Nodes use PDUs to establish, maintain, and clear virtual channel connections (VCC) and virtual path connections (VPC) across the interface. The default address of the

signaling VCC in the connection map address space is VPI.VCI 0.5. You can reconfigure this address to any virtual channel identifier (VCI) under VPI=0, provided the VCI is within the connection map address space. You must always define the signaling VCC under VPI=0. Multiservice Switch nodes support the following signaling protocols according to ATM Forum specifications:

- UNI versions 3.0, 3.1, and 4.0
- IISP version 1.0 (based on UNI signaling versions 3.0 or 3.1)
- ATM inter-network interface (AINI) specification (version 1.0)
- PNNI version 1.0 (a subset of the *Private Network-Network Interface (PNNI) Specification Version 1.0*)
- integrated local management interface (ILMI) version 4.0

Signaling behavior of spared ATM interfaces on Multiservice Switch nodes

For Nortel Multiservice Switch 15000 and Multiservice Switch 20000, the ATM signaling application can be spared as a hot standby application.

Hot standby applications and features offer hitless services during an equipment switchover. Hot standby applications and features incur minimal traffic interruptions and established connections stay up.

Equipment switchovers can occur because:

- an active card fails
- a hitless software migration is in progress

The behavior of ATM signaling is described in Behavior of Multiservice Switch node switched ATM services during an FP switchover (page 30).

See also NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for a full description of hitless services and hot, warm and cold standby applications and features.

Connection map address assignment for ATM IP function processors

Address assignment for ATM IP function processors and 8-port CQC function processors is dynamic, and does not require connection map configuration.

The entire VPI.VCI space is available for connections that these function processors support. The VCI can range from 32 to 65 535. The only limitation is the number of bits for the VPI. If you configure the port with 12 active VPI bits, the VPI can range from 0 to 4095. If you configure the port with eight active VPI bits, the VPI can range from 0 to 255. You configure this value through the *maxVpiBits* attribute under the *AtmIf* component.

The regions of the VPI.VCI space in which the network can allocate switched connections conforms to the ATM Forum ILMI Specification Version 4.0. VPI.VCI connection space configuration has these characteristics:

- SVP connections use VPIs from the range specified in the *minAutoSelectedVpi* and the *maxAutoSelectedVpi* attributes (excluding the range of non-zero VPIs that you configure for VCCs). These attributes are under the *ConnectionAdministrator* component.
- SVC connections use
  - VPIs from 0 and the *firstNonZeroVpiForVccs* attribute to the *firstNonZeroVpiForVccs* attribute plus the *numNonZeroVpisForVccs* attribute less 1. These attributes are under the *ConnectionMapping* component.
  - VCIs for VPIs equal to zero and non-zero. You configure these VCI values using the *minAutoSelectedVciForVpiZero*, *maxAutoSelectedVciForVpiZero*, *minAutoSelectedVciForNonZeroVpi*, and *maxAutoSelectedVciForNonZeroVpi* attributes. These attributes are under the *ConnectionAdministrator* component.

Connection map address assignment for CQC function processors

Nortel Multiservice Switch nodes handle connection map address assignment for CQC function processors in a similar manner for both channels and paths. Nodes handle the VPI.VCI and virtual path identifier (VPI) assignment for a dynamic connection differently for UNIs, IISPs, AINIs than for PNNIs.

When setting up a switched VCC or a VPC on a UNI, IISP, or AINI, the node on the network side of the interface selects an address and assigns that address to the dynamic connection. This address is a VPI.VCI for a VCC and a VPI for a VPC. If the address is available on the interface, the user side accepts the connection. If the address is not available, the network side selects another address and attempts the connection again.

For information on how Multiservice Switch nodes assign address space for 2- and 3-port CQC function processors, see the following sections:

- Network side behavior for SPVC and SVC setup (page 59)
- Network side behavior for SPVP setup (page 59)
- User side behavior for connection and path setup (page 59)

For information on 8-port CQC and ATM IP function processors, see Connection map address assignment for ATM IP function processors (page 57).

For information on how to plan and configure the connection map address space, see Connection mapping (page 204).

Network side behavior for SPVC and SVC setup

When a Nortel Multiservice Switch node is on the network side of the interface during SPVC or SVC setup, it selects an address under VPI=0 where the VCI value in the range specified by the *minAutoSelectedVciForVpiZero* and *maxAutoSelectedVciForVpiZero* attributes. This behavior lets the service provider reserve a range of addresses for dynamic connections under VPI=0. The *minAutoSelectedVciForVpiZero* and *minAutoSelectVciForNonZeroVpi* attributes perform different functions.

In most cases, Multiservice Switch nodes do not use non-zero VPIs for SVCs over UNIs, IISPs, and AINIs. Only if there are no VCIs available under VPI=0 do nodes use values in the *minAutoSelectedVciForNonZeroVpi* and *maxAutoSelectedVciForNonZeroVpi* attributes to search for VCIs under non-zero VPIs in the connection space. Nodes also use these attributes for PORS connections. For more information on PORS connections, see NN10600-435 *Nortel Multiservice Switch 7400/15000/20000 Operations: Path-Oriented Routing System*.

Network side behavior for SPVP setup

When a Nortel Multiservice Switch node is on the network side of the interface during SPVP setup, it selects a VPI value in the range specified by the *minAutoSelectedVpi* and *maxAutoSelectedVpi* attributes.

User side behavior for connection and path setup

When a Nortel Multiservice Switch node is on the user side of the interface, it accepts the connection for the VPI.VCI address for a connection or the VPI address for a path as long as the following conditions exist:

- The address does not support an existing connection or path.
- The address does not violate the address range in the connection map address space.

VPI.VCI assignment for PNNIs

PNNI introduces the concept of a VPCI.VCI address. In a VPCI.VCI address, the node internally maps the virtual path connection identifier (VPCI) that the PNNI protocol uses to the VPI value in the connection map address space. In this release of the Nortel Multiservice Switch node PNNI implementation, each non-associated signaling virtual channel controls only a single interface at the PNNI. The VPCI and VPI have the same numerical value.

When you set up a switched VCC or switched VPC on a PNNI, the behavior of a Multiservice Switch node depends on its position as either a preceding side or succeeding side node. The following section describe this behavior:

- Multiservice Switch node as preceding side (page 60)
- Multiservice Switch node as succeeding side (page 60)

The figure Example of VPI.VCI assignment for PNNI networking (page 62) shows a simple signaling scenario.

Multiservice Switch node as preceding side

When the Nortel Multiservice Switch node is the preceding side node and has the higher node identifier, the node behaves as follows:

- For an SPVC or SVC, the node selects a VPI value of zero and a VCI value in the range specified by the *minAutoSelectedVciForVpiZero* and *maxAutoSelectedVciForVpiZero* attributes.
- For an SPVC or SVC, if no VCIs are available under VPI=0, then the node selects an address in the next VPI in the connection map and a VCI value in the range specified by the *minAutoSelectedVciForNonZeroVpi* and *maxAutoSelectedVciForNonZeroVpi* attributes.
- For an SPVP, the node selects a VPI value in the range specified by the *minAutoSelectedVpi* and *maxAutoSelectedVpi* attributes.

You configure the node identifier on a node through the *nodeId* attribute under the *AtmRouting Pnni ConfiguredNode* component.

Multiservice Switch nodes also generate the connection identifier information element (IE), with the selected VPI.VCI (for SPVCs and SVCs) or VPI (for SPVPs or SVPs) and inserts this information in the call setup request.

When the node is the preceding side node for any connection or path setup request and has the lower node identifier, it does not participate in the VPI.VCI (for SPVCs and SVCs) or VPI (for SPVPs or SVPs) selection. The preceding side node allows the succeeding side node to assign the VPI.VCI or VPI. In response to the call setup request, the preceding side node extracts this information from the connection identifier IE of the first message from the succeeding side node.

Multiservice Switch node as succeeding side

When a Nortel Multiservice Switch node is the succeeding side node and has the higher node identifier, it allocates any VPI value for the connection, provided the connection map address does not support another connection and the VPI value does not violate the configured connection map for the ATM interface. For unspecified VPI.VCI assignments, the Multiservice Switch node selects a VPI value of zero and an unused VCI value in the range specified by

the *minAutoSelectedVciForVpiZero* and *maxAutoSelectedVciForVpiZero* attributes and passes the VPI.VCI information back to the preceding side node in the call proceeding message.

If no VCIs are available under VPI=0, then the node selects an address in the next VPI in the connection map and a VCI value in the range specified by the *minAutoSelectedVciForNonZeroVpi* and *maxAutoSelectedVciForNonZeroVpi* attributes.

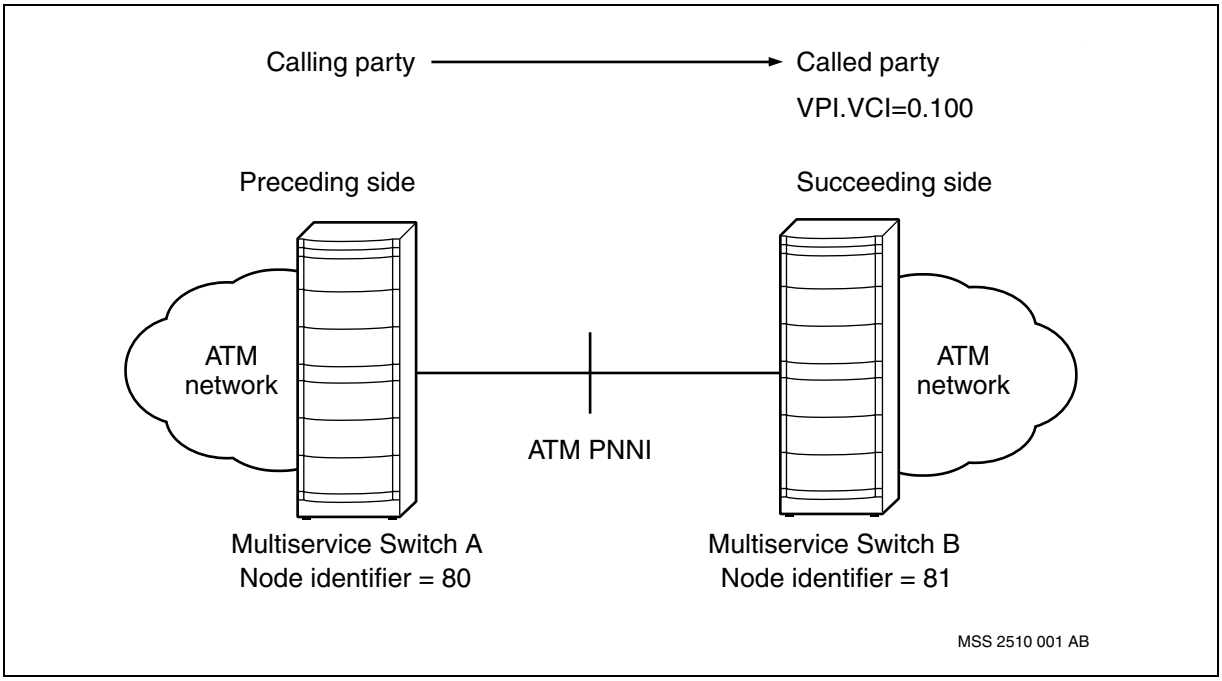
When the node is the succeeding side node and has the higher node identifier for an SPVP, it selects a VPI value in the range specified by the *minAutoSelectedVpi* and *maxAutoSelectedVpi* attributes.

When the node is the succeeding side node and has the lower node identifier, it accepts any VPI.VCI (for SPVCs and SVCs) or VPI (for SPVPs or SVPs) provided that address does not support another connection and does not violate the configured connection map for the ATM interface.

The signaling process has the following characteristics:

- Multiservice Switch A (the preceding side) initiates a call and sends an outgoing call setup request to Multiservice Switch B (the succeeding side).
- Multiservice Switch B has a higher node identifier (81) than Multiservice Switch A (80).
- For a virtual connection setup, Node B selects a VPI.VCI address with VPI = 0 and a VCI value in the range specified by the *minAutoSelectedVciForVpiZero* and *maxAutoSelectedVciForVpiZero* attributes.
- Multiservice Switch B performs all VPI.VCI allocations for all call setups across the PNNI interface between these two nodes.
- For virtual path setup, Multiservice Switch B selects a VPI address where the VPI value in the range specified by the *minAutoSelectedVpi* and *maxAutoSelectedVpi* attributes.
- Multiservice Switch B generates a call proceeding message that includes the selected VPI.VCI or VPI address and sends this message to Multiservice Switch A.

Example of VPI.VCI assignment for PNNI networking



Connection mapping and virtual interfaces

Considerations for the connection map affect signaling requirements for virtual interfaces. The emphasis is on ensuring that connection identification is consistent end-to-end, and that virtual and non-associated signaling can co-exist. For information on connection mapping and virtual interfaces, see the following sections:

- VPI-VPCI mapping for signaling (page 62)
- Resolution of the VCI space (page 65)
- Call setup generating VCIs under VPTs (page 66)

VPI-VPCI mapping for signaling

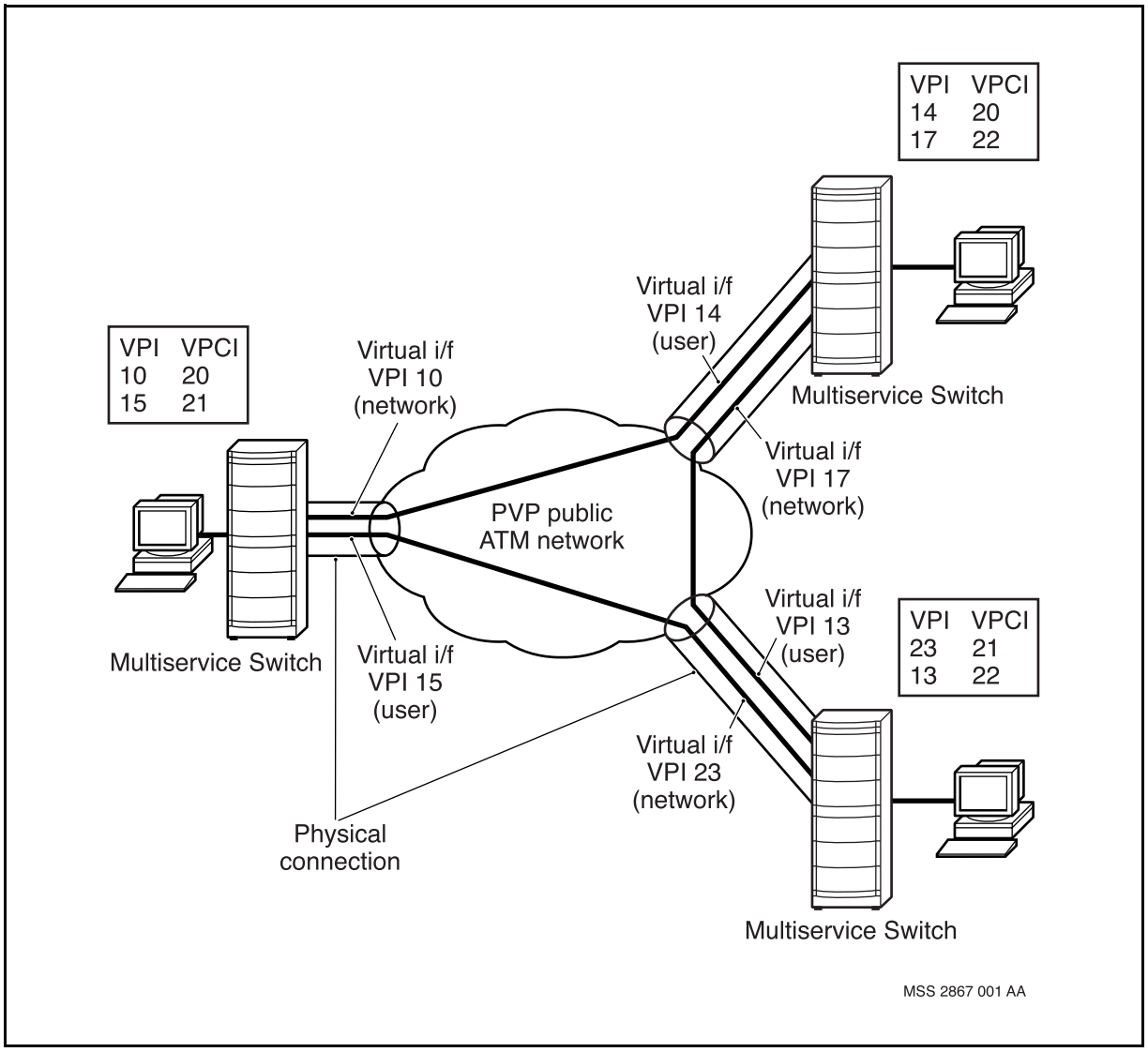
In non-associated signaling, the VPI has the same value as the VPCI. However, when tunneling across a public PVP network, the VPI is in most instances not the same as the VPCI, and the VCI has local significance only. The two ends points of the signaling connection do not share a common identifier. As a result, the nodes must map the local VPI for the VPT to the VPCI of the virtual interface under the VPT. Both end points must have the same VPCI. On each node, configure the VPCI under the virtual interface component, which exists as a subcomponent of the *Vpt* component whose instance is the local VPI.

The figure VPI-VPCI mapping: virtual interface to virtual interface (page 64) shows an example of VPI-VPCI mappings in a network where peer virtual interfaces connect across a VP backbone. The figure VPI-VPCI mapping: virtual interface to real UNI (page 65) shows an example of VPI-VPCI mappings in a network in which Nortel Multiservice Switch node virtual interfaces connect to UNIs on third-party ATM equipment through a VP multiplexer.

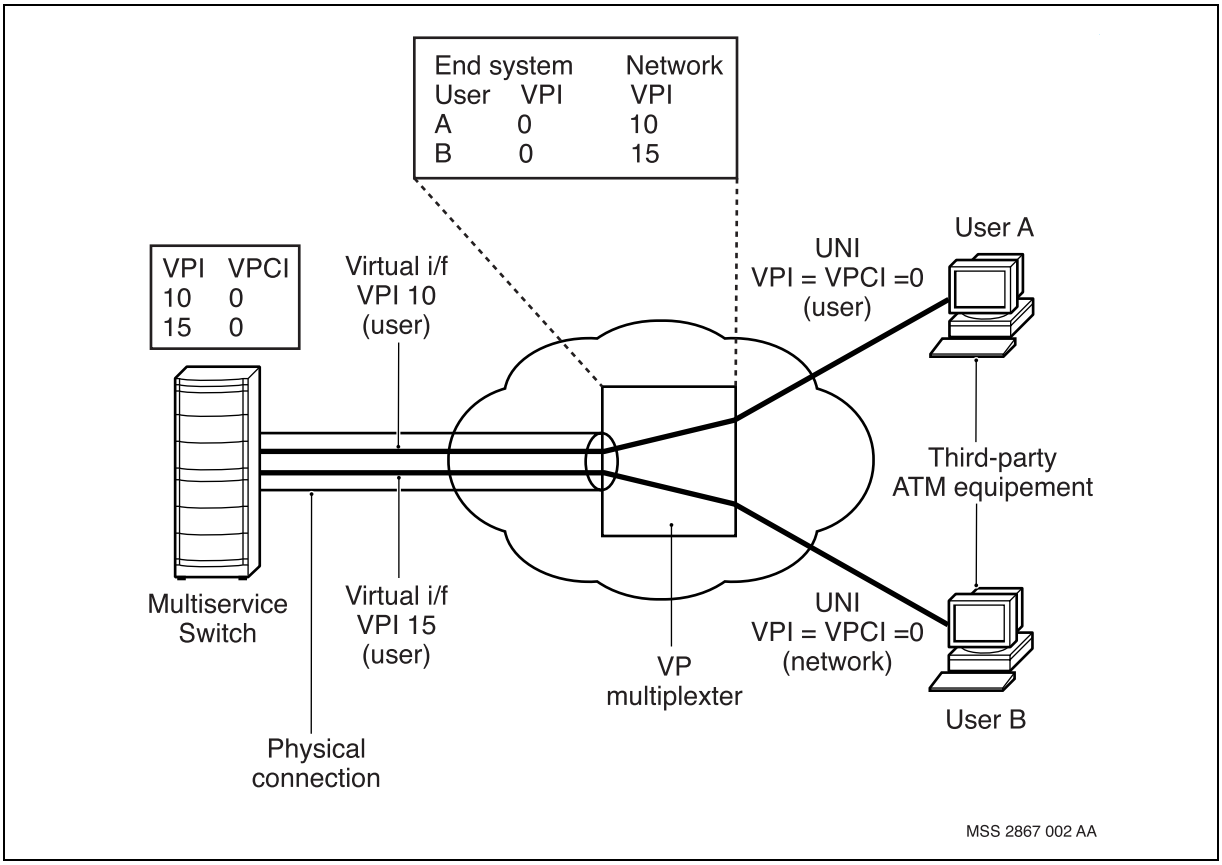
On the network side of a virtual UNI, IISP, or AINI and on the side of a virtual PNNI interface with the higher node identifier, the Multiservice Switch node requests creation of a VCC with the VPI of the associated VPT and any VCI. The VCC identifier in the resulting signaling message consists of the returned VCI and the VPCI of the virtual interface (rather than the VPI associated with the VPT).

If a remote signaling peer chooses a VPCI which is not mapped, on the local side, to the VPI of the virtual UNI, IISP, AINI or PNNI, the local side causes the call to be cleared.

VPI-VPCI mapping: virtual interface to virtual interface



VPI-VPCI mapping: virtual interface to real UNI



Resolution of the VCI space

The signaling channel of a virtual UNI, IISP, AINI, or PNNI interface establishes VCCs under the associated VPT only, using the VPI of the VPT. The non-associated signaling channel establishes VCCs using VPI=0 and VPIs not in use by any VPC or VPT. The coexistence of virtual and non-associated signaling on the same physical interface requires that no VPT or VPC associates with VPI=0. If a VPT or VPC maps to VPI=0, the node disables non-associated signaling.

For an SPVC that you configure under a VPT, a virtual interface within the VPT provides signaling. If there is no virtual interface component under the VPT, the node cannot set up the SPVC. Similarly on the destination side, for an SPVC that terminates under a VPT, a virtual interface within the VPT must provide signaling. As a consequence, you must configure the *calledVpiVci* attribute under the *SrcPvc* component with the VPI of the signaling channel belonging to the terminating virtual interface. Otherwise, the destination node rejects the call setup.

You can set up a PVC under a VPT without a virtual interface being present. This requirement is consistent with all PVCs under an ATM interface.

Call setup generating VCIs under VPTs

If networking provides a wild card for a VCI, then the node allocates a VCI starting with the value that you configure from the *minAutoSelectedVciForNonZeroVpi* attribute under the *AtmIf ConnectionAdministrator* component. The node allocates VCIs and increments the value each time the network specifies a wild card.

On ATM IP function processors, the available VCI values include all values specified by the minimum and maximum auto selected VCIs for the given VPI (range of 32...65535). On CQC function processors, the available VCI values include all values specified by the minimum and maximum auto selected VCIs for the given VPI (range of 0...connmap defined maximum).

Signaling and ILMI

The signaling channel state for UNIs does not depend on the ILMI channel state. The Nortel Multiservice Switch node enables the signaling channel if the node at the other end of the link supports UNIs. If the ILMI channel is down (for example, the far-end node does not support ILMI), the Multiservice Switch node does not disable the signaling channel or clear all associated SVCs. Refer to the *Integrated Local Management Interface (ILMI) Specification Version 4.0* (af-ilmi-0065.000), ATM Forum Technical Committee, 1996 for more information.

Attention: If an ATM device does not support ILMI, disable ILMI for the interface to which it connects.

For information on ILMI addressing, see Address registration through ILMI (page 44).

UNI and ILMI operating versions

The version of ILMI operating at an ATM interface is determined by the configured UNI signalling version under the UNI for the ATM interface. UNI versions 3.x use pre-ILMI 4.0; UNI version 4.0 uses ILMI 4.0.

Pre-ILMI 4.0 does not support ATM group addresses. Semantic checks ensure that group addresses are not configured. ILMI 4.0 defines several new management information base (MIB) objects, which are summarized in the table MIB objects for ILMI 4.0 (page 67). Nortel Multiservice Switch nodes configured for UNI versions 3.x (pre-ILMI 4.0) respond with “noSuchName” to any simple network management protocol (SNMP) Get/GetNext requests that query the MIB objects in the table MIB objects for ILMI 4.0 (page 67).

A node that is upgraded to UNI 4.0 automatically enhances existing pre-ILMI 4.0 capabilities to include support for MIB queries. With the new software installed and with a signalling version of 3.x provisioned, ILMI is fully

compliant. Multiservice Switch nodes that are not operating under UNI 4.0-compliant software are not fully ILMI compliant; nodes that operate under pre-ILMI 4.0 software do not support MIB queries.

MIB objects for ILMI 4.0

Group	Objects
Layer	atmfAtmLayerDeviceType
	atmfAtmLayerIlmiVersion
	atmfAtmLayerNniSigVersion
	atmfAtmLayerMaxSvpcVpi
	atmfAtmLayerMaxSvccVpi
	atmfAtmLayerMinSvccVpi
Virtual Path	atmfVpcBestEffortIndicator
	atmfVpcServiceCategory
Virtual Channel	atmfVccBestEffortIndicator
	atmfVccTransmitFrameDiscard
	atmfVccReceiveFrameDiscard
	atmfVccServiceCategory

MIBs for ILMI 4.0

Support for these requests permits an IME to obtain the values of the following groups of attributes from its peer IME:

- per-system attributes
- per-ATM layer interface attributes
- per-virtual path Attributes
- per-virtual channel attributes
- address registration attributes

Deprecated MIB objects

In general, deprecated MIB objects support backward compatibility with UNI versions 3.x. At this time, Nortel Multiservice Switch nodes do not support deprecated MIB objects and returns a response of “noSuchName” for Get/GetNext requests on the following objects:

- per-physical interface attributes (all objects except Adjacency Information)
- per-ATM layer interface statistics (all objects)

- per-virtual path attributes (*atmfVpcTransmitQoSClass* and *atmfVpcReceiveQoSClass* objects only)
- per-virtual channel attributes (*atmfVccTransmitQoSClass* and *atmfVccReceiveQoSClass* objects only)

ABR exceptions

Because Nortel Multiservice Switch nodes do not support available bit rate (ABR) signalling, nodes return a response of “noSuchName” for Get/GetNext requests on the following objects:

- per-virtual path ABR attributes
- per-virtual channel ABR attributes

Because the ABR MIB groups are conditionally required, Multiservice Switch nodes are compliant to the ATM Forum ILMI 4.0 specification.

Signaling version interworking

Nortel Multiservice Switch nodes support signaling version interworking functions. These functions allow multiple signaling versions to coexist within a Multiservice Switch ATM network. Interworking allows the network to transport signaling PDUs from a UNI interface over a PNNI signaling channel.

Multiservice Switch nodes provide signaling version interworking between the following combinations of protocols:

- UNI 3.1 and PNNI 1.0 signaling version interworking (page 69)
- UNI 3.0 and UNI 3.1 signaling version interworking (page 71), AAL parameter for AAL5 only
- UNI 3.0 and PNNI 1.0 signaling version interworking (page 74), AAL parameter for AAL5 only
- UNI 3.0 and UNI 4.0 signaling version interworking (page 74)
- UNI 3.1 and UNI 4.0 signaling version interworking (page 74)
- UNI 4.0 and PNNI 1.0 signaling version interworking (page 74)

Interworking functions involving UNI 3.0 and UNI 3.1 include the IISP 1.0 protocol.

The signaling version interworking function allows a Multiservice Switch network with a combination of UNI and PNNI signaling interfaces to establish, maintain, and clear ATM SPVCs, SVCs, SPVPs and SVPs.

UNI 3.1 and PNNI 1.0 signaling version interworking

For interworking functions between UNI 3.1 and PNNI 1.0, Nortel Multiservice Switch nodes

- initialize frame discard indication bits in the PNNI 1.0 ATM traffic descriptor IE
- map between UNI 3.1 QoS parameter IE and the PNNI 1.0 extended QoS parameter IE and end-to-end transit delay IE
- map new cause code values in the PNNI 1.0 cause IE to the appropriate cause code values in the UNI 3.1 cause IE

Multiservice Switch nodes apply the appropriate error handling procedures on unrecognized messages and IEs, according to the requirements for the PNNI Version 1.0 and UNI Version 3.1 signaling protocols. Nodes do not use call control to generate unsolicited status messages. As a result, nodes can release a call instead of generating an unsolicited status in response to the received signaling PDU.

For more information on UNI 3.1 and PNNI 1.0 signaling version interworking, see the following sections:

- UNI 3.1 and IISP 1.0v3.1 to PNNI 1.0 signaling interworking (page 69)
- PNNI 1.0 to UNI 3.1 and IISP 1.0v3.1 signaling interworking (page 70)

UNI 3.1 and IISP 1.0v3.1 to PNNI 1.0 signaling interworking

Mandatory requirements in PNNI 1.0 require that call setup messages include the extended QoS parameter IE and the end-to-end transit delay IE. If the originating interface includes these IEs in the call setup message, they are also in the connect message. For more information on generating end-to-end transit delay IEs and extended QoS parameter IEs, see the following documents:

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*
- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

Nortel Multiservice Switch nodes initialize the frame discard bits in the ATM traffic descriptor IE of the call setup and connect messages (if the AAL parameter IE is present). Multiservice Switch nodes base this initialization on the AAL type in the AAL parameter IE. For AAL5 connections, nodes initialize

the frame discard bit to the frame discard allowed cause code for both forward and backward directions. For all other AAL connections, nodes initialize frame discard to the no frame discard allowed cause code for both forward and backward directions.

The PNNI Version 1.0 protocol recognizes all message types and IEs in the UNI Version 3.1 protocol. However, for any unrecognized message types and IEs that a PNNI 1.0 signaling interface receives, nodes apply standard error handling procedures, and uses the message type or IE instruction fields as a basis for action.

PNNI 1.0 to UNI 3.1 and IISP 1.0v3.1 signaling interworking

A PNNI 1.0 call setup message can include the extended QoS IE and end-to-end transit delay IE. When Nortel Multiservice Switch nodes receive messages that include these IEs on a UNI 3.1 or IISP 1.0v31 interface, it removes the extended QoS IE and end-to-end transit delay IE. Multiservice Switch nodes do not map or generate QoS parameter IEs. If the QoS parameter IE is not present in the PNNI Version 1.0 call setup message, nodes clear the call according to UNI Version 3.1 signaling protocol.

PNNI Version 1.0 call setup and connect messages can define frame discard as frame discard allowed in the forward and backward direction. Nodes map this indicator to (00) to comply with the UNI Version 3.1 ATM traffic descriptor IE.

For more information on partial packet discard for UNI Version 3.1 and PNNI Version 1.0, see NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*.

PNNI Version 1.0 introduces new cause codes that nodes map to recognized cause codes for UNI Version 3.1. The table PNNI 1.0 to UNI 3.1 cause IE mapping (page 70) summarizes this mapping. PNNI Version 1.0 also introduces new messages and IEs. The UNI 3.1 signaling protocol applies the proper error handling procedures for unrecognized IEs and messages based on the message type or IE instruction fields.

PNNI 1.0 to UNI 3.1 cause IE mapping

PNNI 1.0 code	Meaning in PNNI 1.0	UNI 3.1 code	Meaning in UNI 3.1
34	Requested called party soft PVPC/PVCC not available	31	Normal, unspecified
		(1 of 2)	

PNNI 1.0 to UNI 3.1 cause IE mapping (continued)

PNNI 1.0 code	Meaning in PNNI 1.0	UNI 3.1 code	Meaning in UNI 3.1
50	Requested facility not subscribed	31	Normal, unspecified
53	Call clear due to change in PGL	31	Normal, unspecified
(2 of 2)			

UNI 3.0 and UNI 3.1 signaling version interworking

For interworking functions between UNI 3.0 and UNI 3.1, Nortel Multiservice Switch nodes support AAL parameter IEs for AAL5. For UNI 3.0 and UNI 3.1 signaling interworking, Multiservice Switch nodes do not support call establishment requests with an AAL parameter IE of anything other than AAL5.

For more information on UNI 3.0 and UNI 3.1 signaling version interworking, see the following sections:

- UNI 3.0 to UNI 3.1 signaling interworking (page 71)
- UNI 3.1 to UNI 3.0 signaling interworking (page 72)

UNI 3.0 to UNI 3.1 signaling interworking

UNI Version 3.1 does not support the UNI 3.0 AAL parameter IE for mode selection (message or streaming mode). In UNI 3.1, ATM traffic descriptors include both the user traffic and end-to-end F5 OAM cell flow. Nortel Multiservice Switch nodes use the F5 OAM cell for segment or end-to-end (VC terminator) management at the VC level. In UNI 3.0, the protocol does not include the F5 flow traffic in the ATM traffic descriptors.

Multiservice Switch nodes use the F5 flow in VC management. As a result, for interworking in the direction of UNI 3.0 to UNI 3.1, you must take into consideration that the rate drops by 1 cell/s. This variance is very small, but can have an impact on configured SCR and PCR values in situations where optimized performance is in place.

The table UNI 3.0 to UNI 3.1 cause IE mapping (page 72) summarizes cause code mappings (octet 6) from UNI Version 3.0 to UNI Version 3.1.

The table UNI 3.0 to UNI 3.1 QoS IE mapping (page 72) summarizes the coding standard changes to the QoS IE.

UNI 3.0 to UNI 3.1 cause IE mapping

UNI 3.0 code	Meaning in UNI 3.0	UNI 3.1 code	Meaning in UNI 3.1
10	VPCI/VCI unacceptable	36	VPCI/VCI assignment failure
51	User cell rate not available	37	User cell rate not available
93	AAL parameters can not be supported	78	AAL parameters cannot be supported

UNI 3.0 to UNI 3.1 QoS IE mapping

Field and octet	Meaning in UNI 3.0	Meaning in UNI 3.1
Coding standard (octet 2) QoS (octets 5 and 6)	Unspecified QoS	ITU-T, Unspecified QoS
Coding standard (octet 2) QoS (octets 5 and 6)	Not supported	ITU-T, Reserved QoS parameter use

UNI 3.1 to UNI 3.0 signaling interworking

The table UNI 3.1 to UNI 3.0 AAL parameter IE mapping (page 72) shows how the interworking function maps the AAL parameter IE.

In UNI 3.1, octet 5a in the broadband bearer capability IE can be present when BCOB-X is stored indicated in octet 5. In UNI 3.0, octet 5a is present if BCOB-X is indicated in octet 5. The table UNI 3.1 to UNI 3.0 BBC IE mapping (page 73) summarizes BBC-IE mapping (octets 5 and 5a).

The table UNI 3.1 to UNI 3.0 QoS IE mapping (page 73) summarizes the coding standard changes to the QoS IE.

The table UNI 3.1 to UNI 3.0 cause IE mapping (page 73) summarizes how Nortel Multiservice Switch nodes map UNI 3.1 cause codes to UNI 3.0 codes (octet 6).

UNI 3.1 to UNI 3.0 AAL parameter IE mapping

Field and octet	Meaning in 3.1	Meaning in 3.0
CPCS-SDU size (Octet 6.1, 6.2)	Forward maximum size is zero	Invalid value (see [1])
CPCS-SDU size (Octet 7.1, 7.2)	Backward maximum size is zero	Invalid value (see [1])
	(1 of 2)	

UNI 3.1 to UNI 3.0 AAL parameter IE mapping (continued)

Field and octet	Meaning in 3.1	Meaning in 3.0
Mode ID (Octet 9)	field not used	Mode identifier
Mode (Octet 9.1)	field not used	Message mode (see [2])
[1] When the IE reaches a UNI 3.0 user, a non-mandatory content error can cause a discarded AAL IE. [2] The field is optional, so an alternative to message mode is to do nothing here. Calling and called end systems can use different modes over the connection. Most current AAL5 implementations use only the message mode.		
(2 of 2)		

UNI 3.1 to UNI 3.0 BBC IE mapping

Meaning in 3.1	Meaning in 3.0
Bit is unspecified	No indication of traffic and time requirements
If octet 5a is not present when Class X or Class VP is indicated in octet 5, construct a UNI 3.0 message with a No Indication value in both the traffic type and the timing requirements fields of octet 5a.	

UNI 3.1 to UNI 3.0 QoS IE mapping

Field and octet	Meaning in UNI 3.1	Meaning in UNI 3.0
Coding standard (octet 2) QoS (octets 5 and 6)	ITU-T, Unspecified QoS	Unspecified QoS
Coding standard (octet 2) QoS (octets 5 and 6)	ITU-T, Reserved QoS parameter use	Not supported

UNI 3.1 to UNI 3.0 cause IE mapping

UNI 3.1 code	Meaning in UNI 3.1	UNI 3.0 code	Meaning in UNI 3.0
16	Normal call clearing	31	Normal, unspecified
36	VPCI/VCI assignment failure	10	VPCI/VCI unacceptable
37	User cell rate not available	51	User cell rate not available
78	AAL parameters cannot be supported	93	AAL parameters cannot be supported

UNI 3.0 and PNNI 1.0 signaling version interworking

Nortel Multiservice Switch nodes support signaling version interworking between UNI 3.0 and PNNI 1.0 for the AAL parameter IE for AAL5. Multiservice Switch nodes do not support the AAL parameter IEs for AAL1 and AAL3/4. Unless the explicit action indicator in the IE instruction field specifies otherwise, nodes clear signaling PDUs with an AAL parameter IE for AAL1 or AAL3/4 with cause code point *access information discarded* or *AAL parameters cannot be supported*.

For signal interworking between UNI 3.0 and PNNI 1.0 messages, nodes complete intermediate interworking with UNI 3.1 before converting the message for the target protocol. See UNI 3.1 and PNNI 1.0 signaling version interworking (page 69) and UNI 3.0 and UNI 3.1 signaling version interworking (page 71).

UNI 3.0 and UNI 4.0 signaling version interworking

UNI 3.0 and UNI 4.0 signaling version interworking involves an intermediate step. UNI 3.0 interworks with PNNI 1.0 and then PNNI 1.0 interworks with UNI 4.0. For information on UNI 3.0 and PNNI 1.0 interworking, see UNI 3.0 and PNNI 1.0 signaling version interworking (page 74). For information on UNI 4.0 and PNNI 1.0 interworking, see UNI 4.0 and PNNI 1.0 signaling version interworking (page 74)

UNI 3.1 and UNI 4.0 signaling version interworking

UNI 3.1 and UNI 4.0 signaling version interworking involves an intermediate step. UNI 3.1 interworks with PNNI 1.0 and then PNNI 1.0 interworks with UNI 4.0. For information on UNI 3.1 and PNNI 1.0 interworking, see UNI 3.1 and PNNI 1.0 signaling version interworking (page 69). For information on UNI 4.0 and PNNI 1.0 interworking, see UNI 4.0 and PNNI 1.0 signaling version interworking (page 74)

UNI 4.0 and PNNI 1.0 signaling version interworking

For UNI Version 4.0 to PNNI Version 1.0 signaling version interworking, the PNNI 1.0 protocol recognizes all message types and IEs in the UNI 4.0 protocol. However, for any unrecognized message types and IEs that a PNNI 1.0 signaling interface receives, Nortel Multiservice Switch nodes apply standard error handling procedures, and uses the message type or IE instruction fields as a basis for action.

For PNNI Version 1.0 to UNI Version 4.0 signaling version interworking, PNNI 1.0 introduces new messages and IEs. The UNI 4.0 signaling protocol applies the proper error handling procedures for unrecognized IEs and messages based on the message type or IE instruction fields.

Interworking between UNI Version 4.0 and PNNI Version 1.0 does not require any cause code mapping or frame discard mappings.

AINI signaling interworking

AINI signaling interworks with PNNI signaling. AINI also supports signaling protocol interworking with UNI 3.0, 3.1, 4.0 and IISP 3.0, 3.1 in a transitive manner by which Nortel Multiservice Switch nodes translate all signaling versions to PNNI. The same restrictions and incompatibilities that exist in the translation of the UNI3.0/3.1 versions to PNNI apply to AINI.

Messages with local significance are not forwarded between an AINI and a PNNI interface and so there is no need to specify an interworking procedure. The following messages are of local significance: Status Enquiry, Status, Restart, Restart Acknowledge, Call Proceeding, Release Complete, Drop Party Acknowledge.

The Release Complete message may carry information of global significance when used as the first call-clearing message. As such, it is translated to the RELEASE message when the call is cleared in the backward direction towards the calling user.

The following PNNI messages are not supported at the AINI: Trace Connection and Trace Connection Acknowledge

In general, messages and information elements are mapped to their equivalent counterparts when converting between AINI and PNNI 1.0, except when noted in the following sections:

- AINI to PNNI 1.0 (page 75)
- PNNI 1.0 to AINI (page 76)

AINI to PNNI 1.0

The format of the crankback information element in PNNI 1.0 is different from that in AINI. As a result, a one-to-one mapping is not possible at the interworking points.

Table Mapping of the crankback IE from AINI to PNNI (page 76) defines the mapping of the crankback IE from AINI to PNNI 1.0. The basic information included in the AINI crankback IE is expanded to include a preceding node identifier containing the PNNI NodeId of the DTL terminator and a succeeding node identifier of all zeroes. On Nortel Multiservice Switch, the port identifier of the blocked link is set to zero. Local routing at the DTL terminator is exhaustive. As a result, the port identifier is not relevant to a PNNI crankback attempt. The mapping is interpreted as a blocked node terminating type and is used by the DTL originator or entry border node to exclude the blocking node as a DTL terminator during alternate routing attempts. The blocking node may still be used as a PNNI transit node. The level of the crankback is set to the lowest level active on the DTL terminator. As a result, it can be interpreted properly in a multi-hierarchical network.

Mapping of the crankback IE from AINI to PNNI

Octect	Crankback IE field	From AINI IE value	To PNNI IE value
5	Crankback level	255	Lowest PNNI level of the DTL terminator
6	Blocked transit type	Call is blocked at or beyond the succeeding node	Blocked link
6.1 to 6.22	Preceding node identifier of the blocked link	Not present	PNNI NodeId of the DTL terminator. This is the last NodeId of the topmost DTL received in the original setup indication from the PNNI network.
6.23 to 6.26	Port identifier of the blocked link.	Not present	The last PNNI PortId of the topmost DTL received in the original setup indication from the PNNI network
6.27 to 6.48	Succeeding node identifier of the blocked link	not present	PNNI NodeId of all zeroes
7	crankback cause	variable depending on circumstance	Crankback cause from the received AINI crankback information element
7.1 and greater	crankback cause diagnostics	not present	omitted

PNNI 1.0 to AINI

The format of the crankback information element in PNNI 1.0 is different from that in AINI. As a result, a one-to-one mapping is not possible at the interworking points.

Table Mapping of the crankback IE from PNNI to AINI (page 77) defines the mapping of the crankback IE from PNNI 1.0 to AINI. The crankback level is set to 255 which is beyond the normal range of levels permitted in a PNNI network (0 to 104) to indicate that the crankback IE was generated outside the scope of the PNNI network. The blocked transit type is always set to a new code point to indicate that the call has blocked at or beyond the succeeding node. The mapping removes the blocked transit identifier because this information is specific to the PNNI domain which generated it. In both translations, no diagnostic information is inserted into the crankback IE. The block transit identifier and the crankback cause diagnostics are omitted because they contain routing information that is not meaningful outside the network in which it is generated.

Mapping of the crankback IE from PNNI to AINI

Octect	Crankback IE field	From PNNI IE value	To AINI IE value
5	Crankback level	Variable	255
6	Blocked transit type	Variable	Call is blocked at or beyond the succeeding node
6.1 to 6.48	Blocked transit identifier	Variable depending on where the call blocked	Omitted.
7	crankback cause	Variable depending on the circumstance	Crankback cause from the received PNNI crankback information element.
7.1 and greater	crankback cause diagnostics	Value depends on the crankback cause value	Omitted

Signaling interworking for SVPs

Some third-party equipment can generate or support SVP call setup under UNI 3.x and IISP 1.0. Nortel Multiservice Switch nodes support these connections under the VP broadband bearer class. For information on allowable parameter combinations, see the allowable parameter combination tables in NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*.

Signaling for point-to-multipoint connections

Nortel Multiservice Switch nodes implement signaling for point-to-multipoint SPVCs and point-to-multipoint SVCs at UNIs, IISPs, AINIs and PNNIs. The following sections describe call establishment, call clearing, and call error handling.

Throughout this discussion of point-to-multipoint signaling, note that for PMP connections originating outside the Multiservice Switch network

- the root can be B-ISDN terminating equipment (for UNI), any terminating equipment (for IISP or AINI), or the preceding side (for PNNI).
- the leaf can be B-ISDN terminating equipment (for UNI), any terminating equipment (for IISP or AINI), or the succeeding side (for PNNI)

See the following documents for complete details on standard implementation requirements:

- *Interim Inter-switch Signaling Protocol (IISP) Specification Version 1.0*, (af-pnni-0026.000), ATM Forum Technical Committee, 1994
- *ATM Inter-Network Interface (AINI) Specification Version 1.0*, (af-cs-0125.000), ATM Forum Technical Committee, 1999

- *Private Network-Network Interface (PNNI) Specification Version 1.0* (af-pnni-0055.000), ATM Forum Technical Committee, 1997
- *User-to-Network Interface Specification Version 3.0* (af-uni-0010.001), ATM Forum Technical Committee, 1993
- *User-to-Network Interface Specification Version 3.1* (af-uni-0010.002), ATM Forum Technical Committee, 1993
- *User-Network Interface Signaling Specification Version 4.0* (af-sig-0061.000), ATM Forum Technical Committee, 1996

For more information on signaling for point-to-multipoint connections, see the following sections:

- Call establishment (page 78)
- Call establishment information flow (page 78)
- Party establishment, re-establishment, and retry for point-to-multipoint SPVCs (page 81)
- Call clearing (page 83)
- Call clearing information flow (page 83)
- Information flow exceptions (page 84)
- Connection maintenance and call clearing (page 85)
- Call error handling (page 85)
- Call clearing at leaf termination for point-to-multipoint SPVCs (page 85)

Call establishment

To set up a point-to-multipoint connection, a Nortel Multiservice Switch root node will establish a connection between itself and the initial leaf. Subsequent parties can then be added to the point-to-multipoint connection. This is accomplished through mechanisms where, in response to add party requests received or generated by the root node, local roots and local leaves are created at branching nodes, as required, with the party ultimately terminating at the remote leaf.

Multiservice Switch nodes process add party requests serially on a per-connection basis. The root node will wait for the current add party request to be completed before issuing a subsequent add party request.

Call establishment information flow

The following nodes can be involved in the initial call establishment:

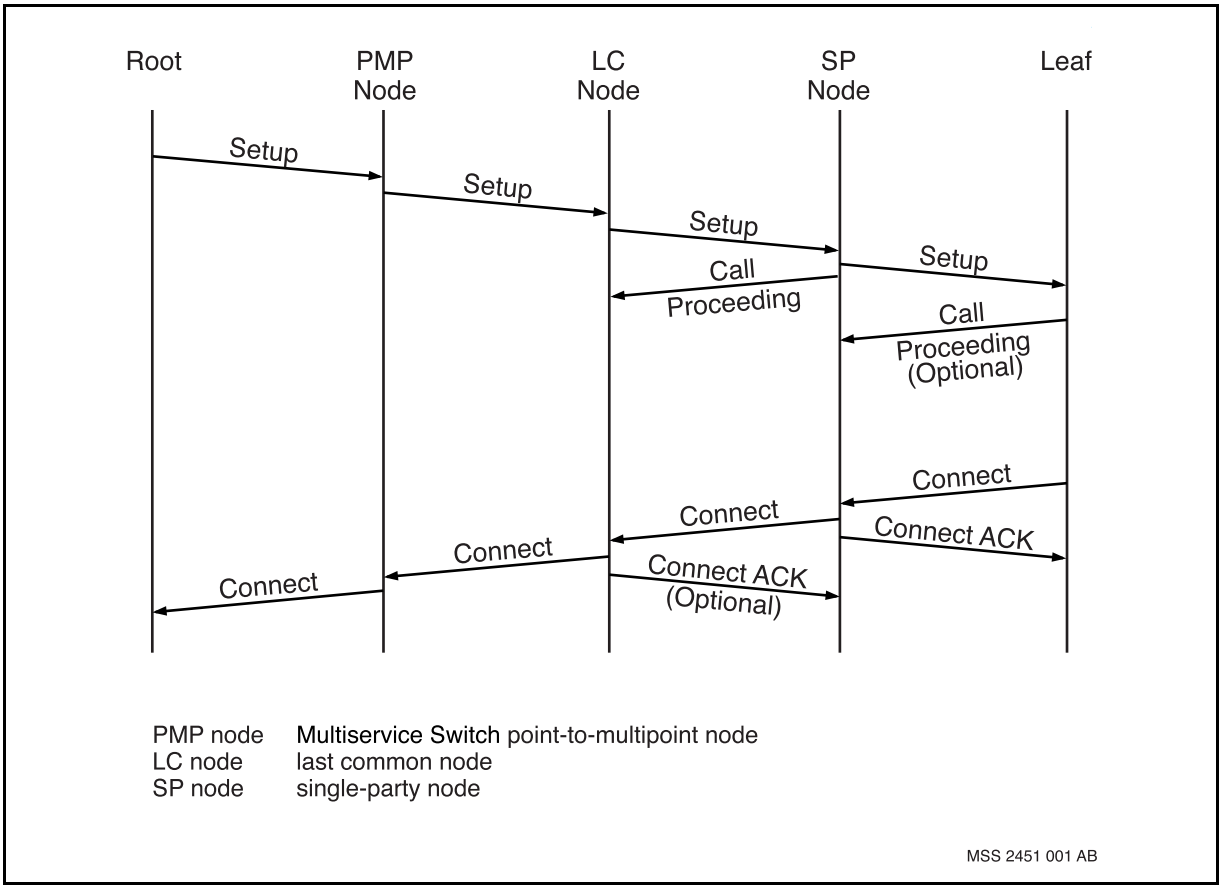
- point-to-multipoint node (PMP node): a network node at which the information flow uses an established connection to a party that participates in a call

- last common node (LC node): a network node at which the information flow uses an established call or connection to a party at the receive interface, and an unused interface for transmission
- single party node (SP node): a network node at which the connection passes information flow to and from a single party of a call

The root sends a SETUP message to the first leaf to establish the initial point-to-point connection. The leaf acknowledges the SETUP message and returns a CONNECT message to the root node.

The figure Add party information flow: connection to initial leaf (page 79) summarizes the information flow that establishes the connection to the initial leaf.

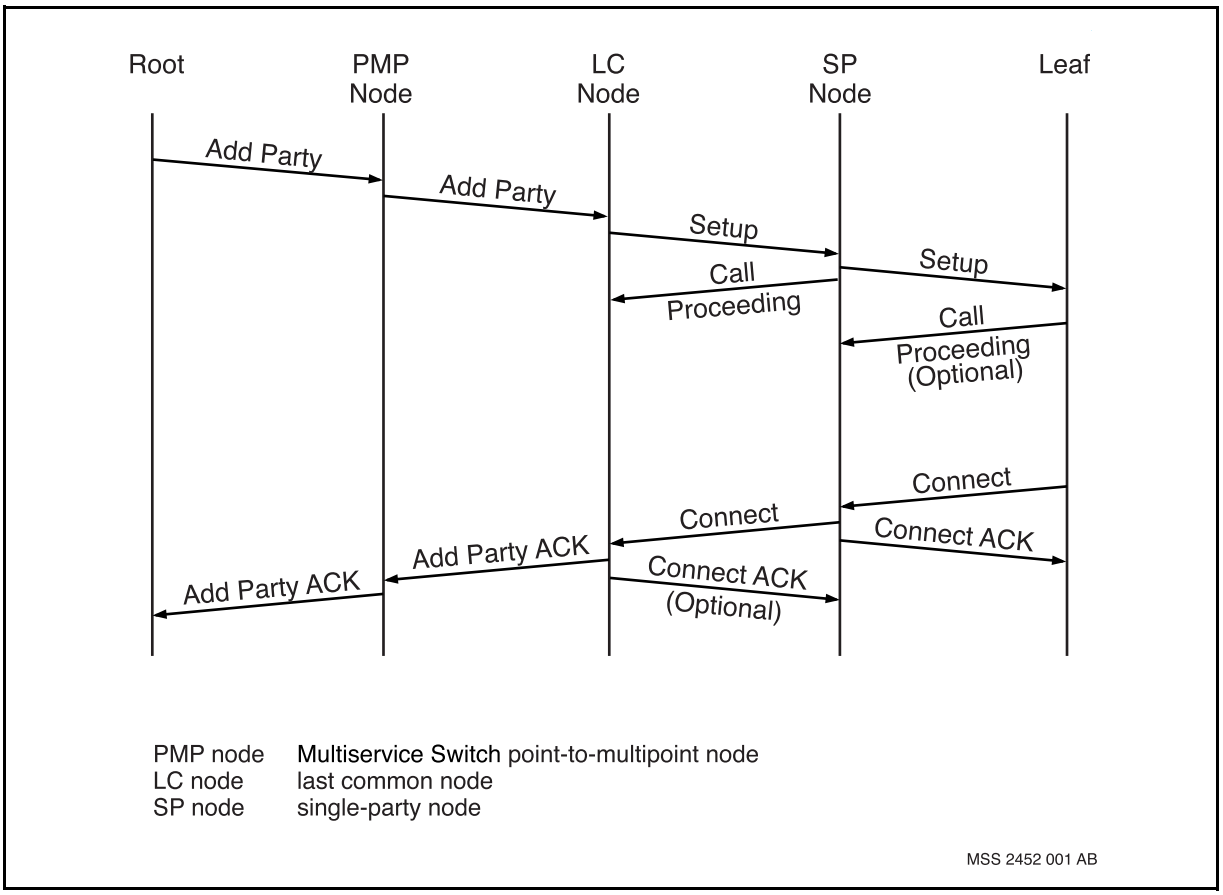
Add party information flow: connection to initial leaf



To add a subsequent leaf, the Nortel Multiservice Switch root node sends an ADD PARTY message to the leaf. The ADD PARTY message can pass through several nodes along an existing VCC prior to branching off to the new leaf. At the LC node (the branch node), the node converts the ADD PARTY message to a SETUP message, which it sends on to the leaf. When the leaf

acknowledges the SETUP message, it returns a CONNECT message to the LC node. The LC node then converts the CONNECT message to an ADD PARTY ACKNOWLEDGE message and forwards it on to the root. The figure Add party information flow: additional leaves (page 80) summarizes the information flow that establishes the connection to a subsequent leaf.

Add party information flow: additional leaves



As the figure Add party information flow: additional leaves (page 80) shows, the LC node converts the ADD PARTY message to a SETUP message in conjunction with the ATM call router. If the ATM call router cannot find an existing branch, it converts the ADD-PARTY message to a SETUP message to create the new branch.

The LC node distinguishes between the CONNECT message from the first leaf on a new branch and all subsequent CONNECT messages received on that branch. For the first leaf, the LC node returns a CONNECT message. For subsequent leaves, the LC node returns an ADD PARTY ACKNOWLEDGE message.

CALL PROCEEDING and CONNECT ACK messages locally acknowledge SETUP and CONNECT messages respectively. ADD PARTY, SETUP, CONNECT and ADD PARTY ACK messages are global.

Party establishment, re-establishment, and retry for point-to-multipoint SPVCs

The point-to-multipoint (PMP) connection is established upon activation of PMP SPVC provisioning. Depending on the arrival sequence of provisioning data, the initial party establishment attempt is made. The party attempt results in one of the following items:

- success (CONNECT or ADD_PARTY ACK message received),
- crankback (RELEASE message received with crankback IE) or
- failure (RELEASE message generated or received without crankback IE).

If the RELEASE message contains a crankback IE then an alternate routing attempt is made immediately through a different path. This alternate routing attempt can result in success, crankback, or failure. Each party attempt must complete (success or failure without crankback) before a subsequent party is attempted. A consequence of this behavior is that the execution thread is broken between attempts which is preferred as it provides other processes and applications with a chance to run and allows topology changes owing to previous party connections to be incorporated into subsequent party route selections. The total number of times a single party is allowed to attempt alternate routing based on reception of crankback is governed by the maximum number of *alternateRoutingAttempts* attribute for point-to-point calls.

Once a party has been attempted (and results in success or failure), the next party establishment attempt is made (based on internal order of reception of provisioning data - the order of parties attempted is not deterministic). Similarly, once this party attempt is completed, the next party is attempted.

There are three categories of party attempts:

- Party establishment
- Party retry
- Party re-establishment

Parties from the three categories queue up in the following two queues for subsequent servicing pending certain waiting periods governed by timers:

- party re-establishment queue (sequences party re-establishment)
- party establishment queue (sequences party establishment and retry)

Parties from the re-establishment queue are serviced first. When the re-establishment queue is empty, parties from the establishment queue are serviced. The queues will continue to be serviced (serially) until they are both empty. Each *Multicastsource* component (*Msrc*) has its own queues, and prioritization is done by each Msrc. Prioritization is not done between Msrc components.The relevant timers associated with the three party attempt categories are as follows:

- hold-off timer (*atmif/n uni|pnni|iisp|aini softPvpAndPvcHoldOffTime* attribute)
- short retry timer (hard coded to 0.5 sec)
- retry timer (*atmif/n uni|pnni|iisp|aini softPvpAndPvcRetryPeriod* attribute)

The establishment model presented and the timer governed waiting periods mirror and are consistent with the current behavior for PTP SPVC/Ps.

Party establishment

Upon reception of provisioning data for a party, a short timer is started (0.5 sec). The purpose of this short timer is to allow all provisioning data associated with the party to be received and processed. Upon expiry of the timer, the individual party is placed on the tail of the party establishment queue which queues up parties ready to establish or to retry.

Party retry

When a party is serviced off the party establishment queue and the party fails to establish (after all crankback attempts), the retry timer is started for the party. When the retry timer expires the party is put back on the tail of the party establishment queue for servicing.

Party re-establishment

When a connected party is released (either due to network or user initiated release), the hold-off timer is started for the party. Upon expiry of the hold-off timer, the party is placed at the tail of the party re-establishment queue.

All parties released

If the entire PMP SPVC connection fails due to network resource failure at a point affecting all leaves (root link for example), upon detection of failure, the PMP SPVC connection and all parties are released. An immediate attempt (following the expiry of the hold-off timer) to re-establish the PMP SPVC is made.

If all *party/m* components are deleted, then the entire PMP SPVC connection is torn down and associated resources are released. Note that a semantic check will ensure that the operator deletes the *msrc* component as well.

Subset of parties released

If a set of specific party connections fails, then the source attempts party connection re-establishment.

If a specific *party/m* component is deleted then the party connection is torn down and associated resources are released.

Call clearing

A Nortel Multiservice Switch node clears a call from a point-to-multipoint connection in response to a request from either the root node or the leaf to be dropped. The node does not support proxy drop-party requests (that is, a request from a leaf other than the leaf that the connection drops). The table Party states for PMP connections (page 84) summarizes recognized party states.

Call clearing information flow

To initiate call clearing, the user sends a RELEASE or a DROP PARTY message to the network.

The node sends a RELEASE message if all parties in the connection are in the process of dropping out or have already dropped out. The network clears all parties on the connection. Parties that are dropping out (drop party initiated and drop party received party states) enter the null party state, in which all timers stop. The network clears parties in add party received state or active state towards the remote user. The network re-establishes parties in the add party initiated state with a new call reference.

The node sends a DROP PARTY message if

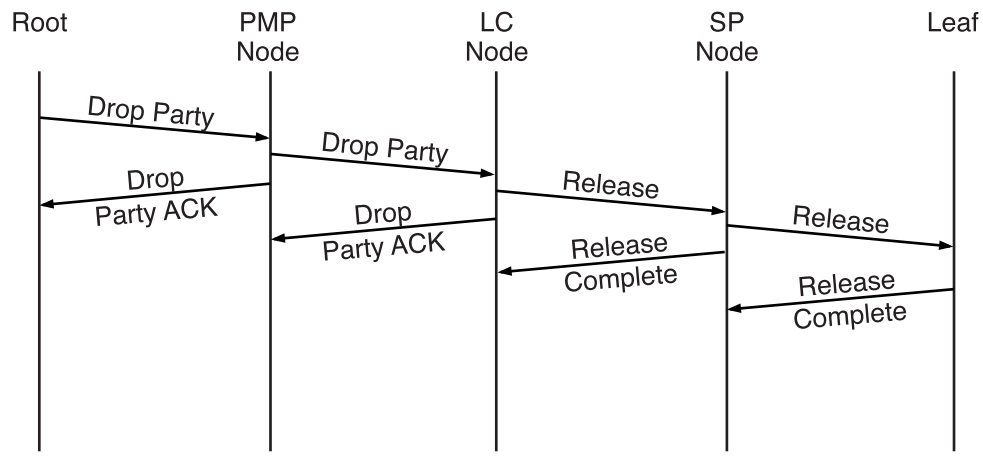
- the party is in the active or add party initiated party states
- other parties in the connection are in the add party initiated, add party received, or the active party states

The DROP PARTY message starts the timer. Normally, the network clears the call before the timer expires and responds with DROP PARTY ACK. When the timer expires, if there is a party in the add party initiated, add party received, or the active states, the node sends a DROP PARTY ACK message into the network. Otherwise the node sends a RELEASE message.

To initiate party clearing, the network nodes send and respond to RELEASE or DROP PARTY messages in a manner that is similar to user-initiated party clearing.

The figure Drop party information flow (page 84) shows a simplified example of drop party. DROP PARTY and RELEASE are global messages that nodes acknowledge locally through DROP PARTY ACK and RELEASE COMPLETE messages respectively.

Drop party information flow



PMP node Multiservice Switch point-to-multipoint node
LC node last common node
SP node single-party node

MSS 2453 001 AB

Party states for PMP connections

Party state	Definition
Null	The party does not exist and there is no end point reference.
Add Party Initiated	A SETUP or an ADD PARTY message was issued.
Add Party Received	A SETUP or an ADD PARTY message was received.
Drop Party Initiated	A DROP PARTY message was issued.
Drop Party Received	A DROP PARTY message was received.
Active	A CONNECT or ADD PARTY ACK message was received.

Information flow exceptions

In response to a SETUP message, the user or network can reject a call or connection (for example, because of the unavailability of a suitable virtual channel). To reject a call or a connection, the user or network

- responds with a RELEASE COMPLETE message, provided no other response has been sent previously
- releases the call reference
- enters the null state

In response to an ADD PARTY message, the user or network can reject an ADD PARTY REQUEST message by responding with an ADD PARTY REJECT message, provided no other response has been sent previously. After sending the ADD PARTY REJECT message, and if no parties remain in the active or add party received states, the root sends a RELEASE message to the leaf. The cause identified in the RELEASE message is #31, Normal unspecified.

When the virtual channel restarts for any reason, the network clears all parties associated with the virtual channel. The network initiates normal drop party procedures toward the leaves for all parties associated with the call. On the interface the party state for all parties associated with the virtual channel is null.

Connection maintenance and call clearing

In the case of point-to-point SVCs and SVPs, the network nodes tear down the connection after the source node clears the call. For point-to-multipoint connections, call clearing procedures take into consideration that multiple parties use the connection. If a party drops off the connection, only the call for that party clears. If other parties are on the same connection in the add party initiated, add party received, or active states, the nodes maintain the connection. Otherwise, the nodes bring the connection down.

Clear collisions that occur during clear party procedures get special consideration. Two types of clear collisions affect point-to-multipoint connections:

- DROP PARTY messages issued simultaneously by the user and the network
- the two last parties at an interface issue clearing messages

To resolve these conditions, the nodes in the network issue DROP PARTY ACKNOWLEDGE, or RELEASE messages.

Call error handling

In general, the Nortel Multiservice Switch node applies point-to-point error handling first. Where point-to-point error handling recovers the connection, the node takes no further action.

If point-to-point error handling does not recover the connection, the node applies point-to-multipoint error handling, which involves the ADD PARTY REJECT, STATUS, and STATUS ENQUIRY messages.

Call clearing at leaf termination for point-to-multipoint SPVCs

As is the case with PTP SPVCs, a PMP SPVC leaf termination does not necessarily represent the final end user destination. The final destination may lie either directly at the user end of the UNI or may lie further beyond.

Additionally, the equipment at the user end of the UNI may or may not be PMP capable. PMP SPVCs and PMP SVCs will progress SETUP and ADD_PARTY messages out the UNI interface.

In the event that an ADD PARTY is received which requires the creation of a PMP Dst PVC leaf (the destination address provided is either the default address of the port or is a provisioned address with an associated TerminateSPvpAndSPvc subcomponent), the ADD PARTY message is converted to a SETUP which is sent towards the egress side and the new PMP Dst PVC leaf is established. Each such occurrence necessitates the creation of a new branch. The capability of terminating multiple leaves of the same PMP call, each with its own vpi/vci on the same port is referred to as logical multicasting at an egress interface.

Generally, if the calledVpiVci contained in a SETUP message received at the destination is in use, then the SETUP is rejected with cause code #34 “Requested called party soft PVPC or PVCC not available”. Additionally, it is not allowed to terminate a leaf of a PMP SPVC on a provisioned Dst Pvc. In this case, the SETUP message is also released towards the source with cause code #34.

However, a special case of the *calledVpiVci* used at the destination needs to be considered. This case can arise as a result of race conditions occurring within an ATM network during SPVC failure recovery. It is possible that, depending upon which link fails in the network, the terminating end of an SPVC (party) can receive a re-establishment SETUP message prior to receiving the RELEASE message associated with the failure. To immediately re-establish the connection segment in this case and to avoid falling back to retry interval re-establishment from the source, a comparison is made between the calling party information in the SETUP message and the operational SPVC destination point. If the SETUP is coming from the exact same originator (*callingAddress*, *callingVpi*, *callingVci* identical with *callingAddress* non-null) then the race condition described previously has been detected. Rather than reject the new SETUP with cause code #34 as previously discussed, the original call is released with cause code #16 “normal call clearing” and the new SETUP message is processed.

Interactions between virtual interfaces and VPTs

Interaction between virtual interfaces and VPTs have the following characteristics:

- Non-associated signaling cannot set up an SVC or SPVC within the VPI space of a VPT or VPC (regardless of the existence of a virtual interface within the VPT).
- A Nortel Multiservice Switch node rejects a connection request from a virtual interface if the requested VPI is not the VPI of the associated VPT.

- On an inverse multiplexing for ATM (IMA) card, if the node takes down a VPT with an associated virtual interface because of bandwidth reallocation, the node also takes down the signaling channel within the VPT.

PNNI path trace

The PNNI path trace is a standard-based diagnostic tool that enables you to determine the logical nodes and links that new point-to-point connections in the process of being established traverse in flat and hierarchical PNNI networks.

Path tracing may be initiated for the purpose of determining paths that new connections might take, or for troubleshooting connection establishment problems.

Path tracing does not require new signaling messages but requires the addition of the trace transit list (TTL) information element in several signaling messages. It also requires additional procedures to the standard PNNI call control procedures.

The trace is initiated at the trace source node, which can be any node along the path within the PNNI domain, but is not necessarily the connection source node. Tracing terminates at the trace destination node, although the connection may progress beyond the trace destination node when it reaches the trace destination interface. This interface is defined by one of three conditions:

- This interface is the connection destination interface (that directly supports the connection called party number).
- The next interface which the connection would traverse for the path trace is not a PNNI interface. For example, UNI, IISP, and AINI are possible interfaces.
- The next interface which the connection would traverse for the path trace is administratively designated by the user as a trace destination interface.

Path Trace has two capabilities:

- test connection
- filter

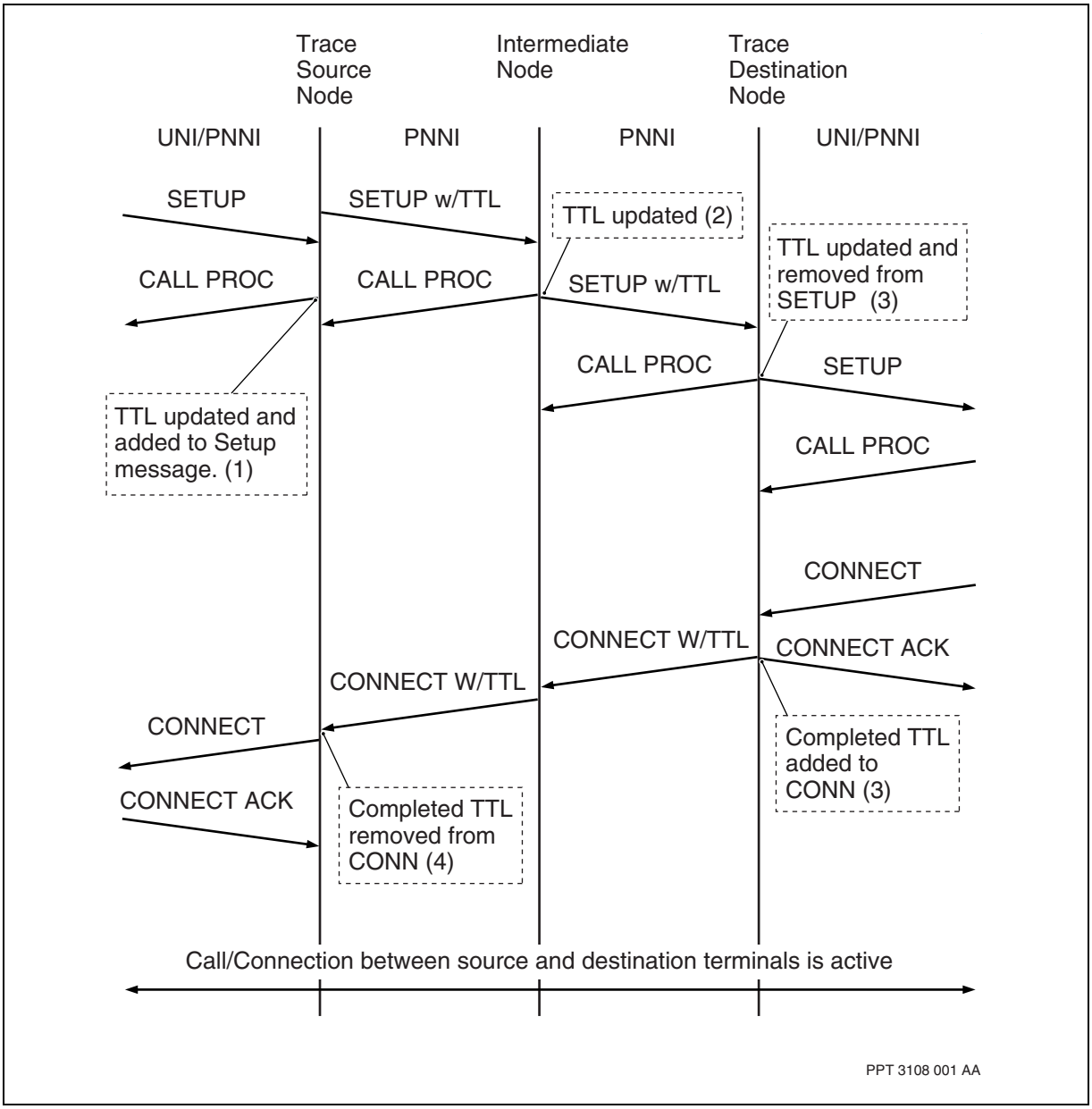
The test connection capability tries to find the route to the destination, acting in a similar manner to the ping command. Test connections can be initiated at the trace source node towards a called party number. A test connection will setup an SPVC from the trace source to the trace destination and will always be released once it reaches the trace destination. The default value to which flag X is set cannot be changed.

The filter capability is provisioned at the trace source node and is used to define the criteria for which a connection will be dynamically traced. The filter criteria includes: called party number, calling party number, connection type, connection cast (point-to-point), and service category. When a filter is provisioned on a certain interface, all connections going through that interface, which satisfy all defined filtering criteria, will be traced. This capability applies to the following connections: SVC, SVP, SPVC, and SPVP connections.

By tracing a connection, the system displays an ordered list of output. This ordered list contains the logical nodeId and portId at the lowest level of the hierarchy that the connection will traverse. This starts with the trace source node and ends at the trace destination node. You can obtain additional information on the trace by setting certain TTL information element flags when initiating the call trace. When multiple filters are activated in different nodes along the call connection path, only the first filter will be able to capture all the trace information. For information on the optional flags for the trace command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*.

The figure Message sequence for the path trace (page 89) illustrates a sample message sequence for the path trace.

Message sequence for the path trace



The following list defines each part of figure Message sequence for the path trace (page 89):

1 The path trace capability is initiated at the trace source node by inserting a trace transit list information element (TTL IE) in the SETUP message. The trace source node adds its trace information including its nodeID and egress interface portID to this TTL IE. Depending on the configuration of the flags for the trace source node, additional trace information can be returned in the TTL IE. This can include crankback information (if the flag

C is set), VPI/VCI (if the flag V is set), and/or call reference values (if the flag A is set). The trace source node then sends the SETUP message on the outgoing interface.

2 All intermediate nodes, which are neither the trace source node nor the trace destination node, perform the procedures that are normally followed for a SETUP message. It completes the trace information and then forwards the message to the next node.

3 At the trace destination node, the trace information is kept so it can be transferred into the corresponding CONNECT message if the connection or party is intended to be left up (flag X is not set, this only applies to path trace filter), or into a RELEASE, RELEASE COMPLETE message if the connection or party is to be immediately cleared (flag X is set as the default value, which applies to the path trace test connection).

4 At the trace source node, the TTL IE is removed from one of the previous messages.

For information on configuring the path trace feature, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

For information on the trace command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*.

PNNI connection trace

The PNNI connection trace enables you to determine the logical nodes and links that existing point-to-point connections traverse in flat and hierarchical PNNI networks.

Connection tracing may be initiated for the purpose of collecting information on these existing connections.

The connection trace requires the following two new signalling messages: TRACE CONNECTION and TRACE CONNECTION ACKNOWLEDGE in addition to the TTL information element when it traces these calls which are already active.

The trace is initiated at the trace source node, which could be any intermediate node along the path of a connection within the PNNI domain, and is not necessarily the connection source node. Tracing terminates at the trace destination node when it reaches the trace destination interface. This interface is defined by one of three conditions:

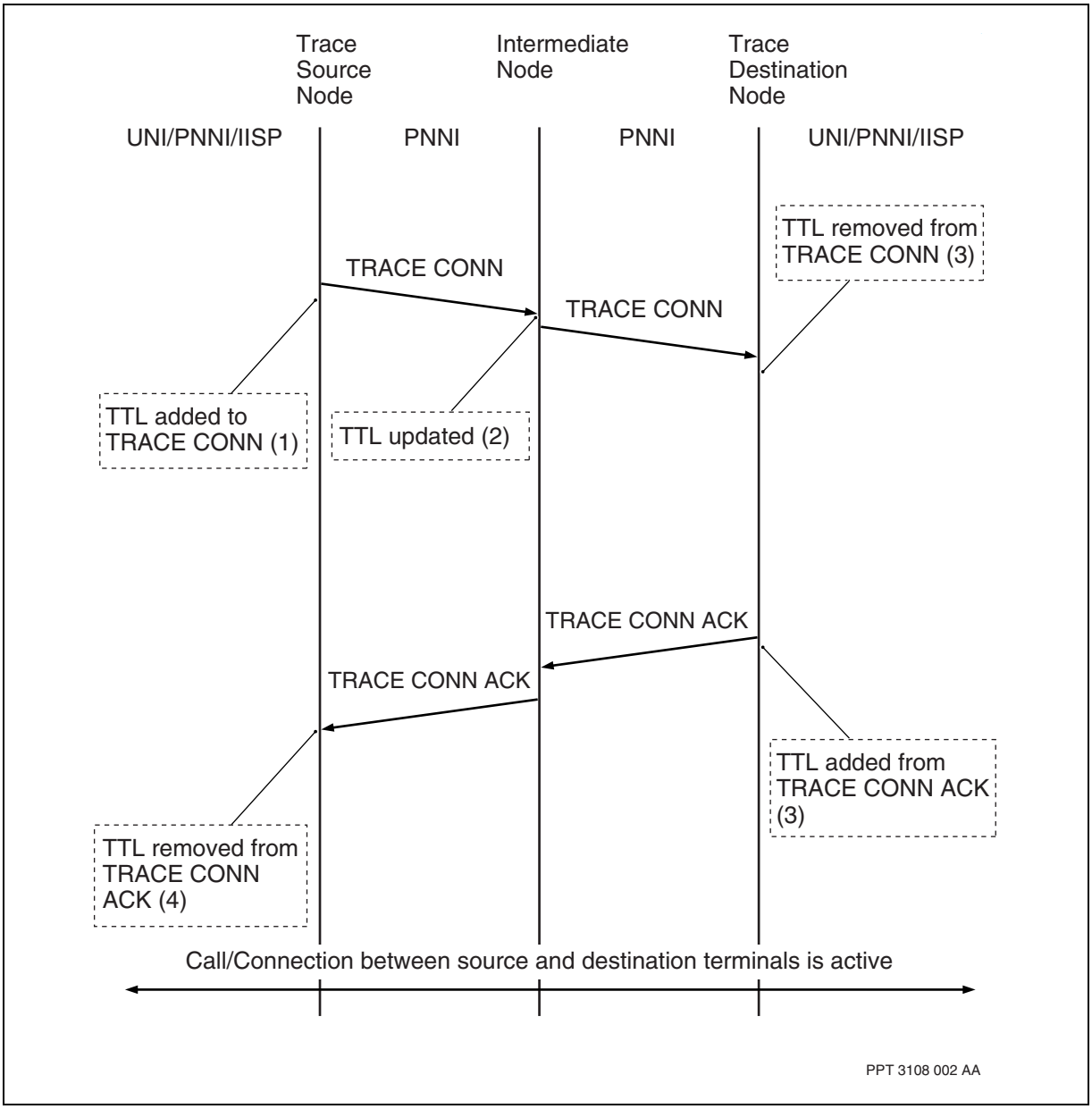
- This interface is the connection destination interface (that directly supports the connection called party number).

- The next interface which the connection traverses for connection trace is not a PNNI interface. For example, UNI, IISP, and AINI are possible interfaces.
- The next interface which the connection traverses for connection trace is administratively designated by the user as a trace destination interface.

By tracing a connection, the system displays an ordered list of output. This ordered list contains the logical nodeId and portId at the lowest level of the hierarchy that the connection has already traversed. This starts with the trace source node and ends at the trace destination node. You can obtain additional information on the trace by setting certain TTL information element flags when initiating the call trace. For information on the optional flags for the trace command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*.

The figure Message sequence for the connection trace (page 92) illustrates a sample message sequence for the connection trace.

Message sequence for the connection trace



The following list defines each of the parts of figure Message sequence for the connection trace (page 92):

1 Connection trace starts at the trace source node for a given connection. The trace source node creates a TRACE CONNECTION message and inserts a TTL IE in the message. Depending on the configuration of the flags at the trace source node, more trace information including VPI/VCI and/or call reference values can be added to the TTL IE.

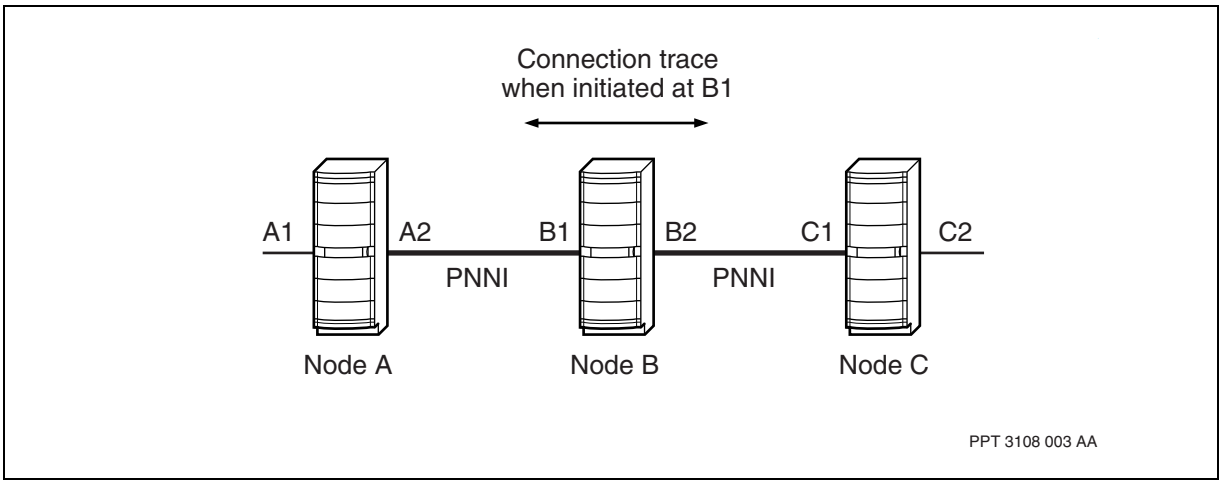
2 All intermediate nodes, which are neither the trace source node nor the trace destination node, complete the trace information and forward the message to the next node

3 The trace destination node completes the trace information and returns the trace information in the TRACE CONNECTION ACKNOWLEDGE message to the trace source node. It should be noted that the flags X and C, used in path trace, are not applicable to connection trace.

4 The trace source node removes the TTL IE from TRACE CONNECTION ACKNOWLEDGE message.

A connection trace may be initiated at any node along the path of a connection. It does not necessarily have to be the connection source or destination node. In such an event, the connection trace would send two TRACE CONNECTION messages, one link side and one switch side providing a complete connection path. The figure Connection trace initiated at the intermediate node (page 93) illustrates the sequence of the connection trace when it is initiated at NodeB. For example, if an established connection originates at interface A1 on NodeA and terminates on interface C2 on NodeC, it goes through the following path: (A1, A2, B1, B2, C1, C2). If a connection trace is initiated for this connection at interface B1, the switch side trace would return (B1, B2, C1, C2) and the link side trace would return (B1, A2, A1) providing a complete connection path.

Connection trace initiated at the intermediate node



For information on configuring the connection trace feature, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

For information on the trace command including the optional TTL information element flags for this command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*.

ATM routing

Nortel Multiservice Switch nodes support two routing approaches:

- static routing over user-to-network interface (UNI), interim inter-switch signaling protocol (IISP) interface, and ATM inter-network interface (AINI)
- dynamic routing (also known as dynamic route discovery) over private network-to-network interfaces (PNNI)

Navigation

Static routing

Nortel Multiservice Switch nodes use a static routing scheme to forward call setup requests across the network. Nodes do not exchange routing information. Routing proceeds according to the *User-to-Network Interface Specification Version 3.1*, *Interim Inter-switch Signaling Protocol (IISP) Specification Version 1.0, and ATM Inter-Network Interface (AINI) Specification Version 1.0*.

As you configure addresses for an interface, and as the node dynamically registers addresses, the Multiservice Switch node enters these addresses into the call routing table. The node also enters the default address for each interface. Each node has one table. The call routing table maps each static and registered address to its corresponding interface. For UNIs, IISPs, or AINIs, the node adds the associated addresses to the call routing table when the interface goes into service, and removes the addresses when the interface goes out of service. You can display the table by listing the *Dna* components under the *AtmRouting* component.

An SVC or SVP cannot exit through the same port on which it entered the Multiservice Switch node. For SPVCs or SPVPs, signaling can use the egress port as configured but routing of the call must follow the above port requirement.

For information on static routing, see the following sections:

- Mechanics of static routing (page 95)
- Process to resolve a tied address match (page 96)

Mechanics of static routing

When a node receives an incoming call setup request (through the signaling channel on a UNI, IISP, or AINI interface), it scans the call routing table for the called address. The called address appears in the call setup request. The node completes a maximal address match by selecting from the call routing table the address with the best match. The node then forwards the call setup request (through the signaling channel) to the next-hop UNI, IISP, or AINI interface associated with the best-match address. The node clears the call if it cannot find an address match.

As a maximal address match example, suppose one IISP interface handles all addresses starting with 393, another IISP handles all addresses starting with 393763, and another IISP handles address 3937632211. The nodes in the network route an incoming call setup request with a called address that starts with 3937632222 to the IISP interface that handles all addresses starting with 393763.

For addresses registered through integrated local management interface (ILMI), Nortel Multiservice Switch nodes enter the first 38 hex digits into the call routing table. The 39th and 40th digits (the selector field) have end system significance only and do not affect the address matching function. For static addresses, Multiservice Switch nodes use all configured digits (up to the maximum of 40) in the address matching function.

SVCs and SVPs cannot exit through the same port on which they entered the Multiservice Switch node. For SPVCs or SPVPs, signaling can use the egress port as configured but routing of the call must follow the above port requirement.

Process to resolve a tied address match

Under UNI, IISP, and AINI, when an address match produces a tie (that is, when more than one interface handles the best-match address), the node that forwards the call selects an interface in a round-robin fashion. The next time the same best-match address tie occurs, the forwarding node selects a different interface. This distribution of match selection improves load sharing when multiple interfaces lead to the same route. This process also helps avoid any route that leads to a failed network element.

The call routing function tries each route associated with a matching address until it establishes a connection. Each address has a primary and an alternate list of interfaces with these characteristics:

- The primary list references the interfaces to their associated primary static addresses, registered addresses, and default addresses.
- The alternate list references the interfaces to their associated alternate static addresses. For more information on configured static addresses, see Static addressing for UNI, IISP, and AINI (page 46).

A Nortel Multiservice Switch node first attempts to find the address in the primary list. If the node cannot establish call setup, the node then attempts to find the address in the alternate list. The starting point in the primary list changes for every new call setup request in round-robin fashion.

Dynamic routing through PNNI

The objective of PNNI is to provide an interface between ATM switches, so that network nodes can construct full-function networks of arbitrary size and complexity. Nortel Multiservice Switch hierarchical PNNI uses topology and reachability aggregation to allow increased stability of networks by

- supporting wider network diameters
- using less memory for the topology database
- improving call setup times

A network must efficiently compute end-to-end QoS-based routes. As network size increases, the network service provider faces the following performance affecting problems:

- increased size of the topology database due to large network size
- increased effort to maintain and update a large topology database
- increased route computation time needed for route selection across large networks
- increased computational complexity for supporting multiple service categories and QoS
- increased bandwidth consumed by routing protocol traffic
- increased database convergence time after topology changes
- increased re-routing upon link or node failures

These scalability issues can be addressed through multi-level PNNI networking (hierarchical PNNI). Within each peer group, nodes exchange topology and link state information so that each node has a detailed topology and link state map of its own peer group. At higher levels, however, detailed information is not necessary. Information about a peer group forms a logical image of the peer group as a single LGN.

Nortel Multiservice Switch nodes fully support all mandatory features described in the ATM Forum's PNNI Version 1.0 routing specification including:

- border node capability
- ability to act as a peer group leader (PGL) at any level in the hierarchy
- ability to represent a peer group as a logical group node (LGN)
- ability to act as a peer to a LGN at the lowest level of the hierarchy

In adherence to this specification, nodes in the PNNI network regularly exchange topology and link state information. In this way, nodes maintain an up-to-date view of the state of the network.

PNNI nodes uses source routing. The first node (source node) in a connection decides the route to the destination physical node or logical group node. If a subsequent node rejects the connection, call setup retraces the path to the originating node. The originating node uses alternate routing information and re-attempts call setup. In a hierarchical PNNI network, entry border nodes along the path expand the DTL stack to reach the next LGN specified in the topmost DTL when it is available. Otherwise, the entry border nodes will expand the stack to reach the called address. At each exit border node, exhausted DTLs are removed from the stack before the call is forwarded to the

next peer group. This source routing approach differs from that used in static routing protocols which require manual configuration for access to addresses across interfaces.

The PNNI node's routing protocol is hierarchical. It provides automatic discovery of network topology and advertisement of link information. Nodes store topology data in a database that the peer group references, where a peer group is a logical collection (grouping) of network nodes at the same level. Nortel Multiservice Switch nodes in a PNNI network use the routing control channel to exchange routing protocol messages.

Attention: When a control processor switch over occurs, all PNNI RCCs restart while signaling channels and all existing connections are unaffected. The RCC also restarts when a critical attribute of the *AtmRouting Pnni* component changes.

Call setup requests enter the network through either a UNI, IISP, or AINI interface at the source node. The network routes the request to peer group nodes based on route computation and connection admission control (CAC) requirements. The network optimizes routes based on administrative weight (AW), end-to-end cell transfer delay (CTD), and end-to-end cell delay variation (CDV). The network routes calls end-to-end from the source node across a sequence of logical group nodes (LGN), where the LGNs are members of the peer group. The peer group leader floods LGN information to the peer group.

For more information on dynamic routing through PNNI, see the following sections:

- Mechanics of PNNI-based dynamic routing (page 99)
- Hello protocol (page 100)
- Topology advertising and synchronization protocols (page 100)
- Peer group elections (page 102)
- Logical group node representation (page 103)
- Communicating information between the PGL and LGN (page 105)
- Topology exchange between lowest level nodes (page 106)
- Topology exchange over logical links (page 107)
- Connection admission control (page 108)
- Optimization criteria (page 109)
- Links (page 109)

For examples on how to build hierarchical PNNI networks, see Hierarchical PNNI examples (page 162).

Mechanics of PNNI-based dynamic routing

Nortel Multiservice Switch nodes can participate fully in a multi-level PNNI (hierarchical PNNI) network. Multiservice Switch nodes can serve as peer group leaders or as border nodes.

When an incoming call setup enters a node through either a UNI, IISP, or AINI interface, the system performs a longest prefix match of the destination address with all of the addresses on the UNIs, IISPs, AINIs and with all of the addresses the PNNI nodes support.

It is possible that the network has several choices with the same best prefix match. Calls that forward according to the PNNI protocol maintain source routing capabilities. Calls that forward according to UNI, IISP, and AINI protocols use static routing. This progression of prefix matching and call setup lets the customer control when the network migrates from static routing to dynamic routing. The PNNI prefix match and call set up rules are as follows:

- For UNI, IISP, and AINI, the best match is chosen.
- If there are multiple PNNIs, then the closest address is chosen.
- For a given length of matching address, a static routing interface is chosen over a PNNI interface.
- If there are multiple best match static interfaces, one of them is chosen.

The route that a node selects through these call setup rules always satisfies the quality of service (QoS) requirements in the call setup request.

If a Multiservice Switch node forwards the call according to the PNNI protocol, it uses the topology and resource information that it acquires from its neighbors to find the best route. The route satisfies the QoS requirements in the call setup request. A Multiservice Switch node also bases route selection on optimization criteria that either the customer or the service provider can define (for example, giving preference to routes with minimal administrative weight).

The originating node uses a generic connection admission control (GCAC) procedure to determine if a route satisfies QoS requirements. GCAC checks each link as the node calculates the route. If the route is suitable, the Multiservice Switch node puts the routing information in the call setup as a designated transit list (DTL). Intermediate nodes along the route forward the call setup based on the route in the DTL. Each intermediate node uses CAC to determine local resource availability. If resources are available, the node forwards the call setup. Otherwise, the node blocks the call setup.

Attention: CAC is sometimes referred to as actual connection admission control (ACAC). Multiservice Switch documentation uses the term CAC for actual CAC and GCAC for generic CAC.

If a node along the route blocks call setup (that is, the call does not reach its destination) the network cranks back the call to the originating node. The originating node either selects a new route or rejects the call. This procedure is a significant improvement over crankback in IISP-based networks. In a PNNI-based network, the originating node determines the entire route (hence the term source routing). As a result, routing loops cannot occur in a network with PNNI-based nodes only. (Routing loops can potentially occur in a poorly planned mixed IISP-PNNI network.)

Hello protocol

In a PNNI network, when a switching system becomes operational, it exchanges Hello messages with neighboring nodes. The objective is to identify if directly connected nodes are within the same peer group and determine the status of the links to those nodes.

Nortel Multiservice Switch nodes use several mechanisms to control the link bandwidth that the Hello protocol consumes. These mechanisms include

- how often Multiservice Switch nodes generate periodic Hello messages
- how frequently Multiservice Switch nodes generate event-driven Hello messages

Hello parameters can be configured

- for the node under the *AtmRouting Pnni* subcomponent
- for a particular level under the *AtmRouting Pnni ConfiguredNode* subcomponent
- for a specific link under the *AtmRouting Pnni RoutingControlChannel* subcomponent

The Hello protocol supports inside link communication between LGNs. For more information on the Hello protocol and other protocols supporting hierarchical PNNI, see Topology exchange over logical links (page 107).

Topology advertising and synchronization protocols

When a function processor receives a Hello message from a neighboring node, the Nortel Multiservice Switch node compares the peer group identifier (PGID) of the neighbor with its own PGID. In this way, the node determines the peer group relationship between the nodes. If the nodes belong to the same peer group, they exchange and maintain link and node information (*twoWayInside*). If the nodes do not belong to the same peer group, they will

share nodal hierarchy information to find a common peer group (this begins at the *twoWayOutside* state). Once a common peer group is found (*commonOutside*), information is exchanged between the highest level LGNs instantiated by the lowest level peer groups.

To describe its local topology, each node builds a PNNI topology state packet (PTSP). The PTSP lists all adjacent links and the link characteristics. The node then intelligently floods (broadcasts) the PTSP message to all adjacent members of the peer group. Each node floods a PTSP message to its link-connected peers, which in turn send the PTSP message to their link-connected peers (assuming the PTSP is newer than the last received version).

Nodes generate and flood PTSP messages to

- periodically refresh the databases of their peers
- update the databases of their peers after a significant change

You use subcomponents under the *Pnni* component to control the following parameters for PTSP messages:

- the link bandwidth that PTSP messages consume (by changing attributes for timer, rate, and interval characteristics)
- how often a Multiservice Switch node generates periodic Hello messages
- how quickly a Multiservice Switch node generates event-driven Hello messages

The PNNI topology advertisement and synchronization protocols let each node in the peer group maintain an identical copy of the topology database.

For Multiservice Switch nodes, the control processor maintains the topology database. The function processor forwards all packets that pertain to topology information to the control processor. The control processor also parses each packet for validation and information contents. For example, if the node has already received a PTSP, the control processor ignores subsequent duplicates. The objective is to minimize the number of updates to the topology database and to limit topology advertising protocol traffic.

The active control processor does not maintain a topology database on the standby control processor. When a control processor switch over occurs, the newly active control processor re-synchronizes with its neighbors to rebuild the topology database.

For more information relating to hierarchical PNNI topology, see Topology exchange between lowest level nodes (page 106) and Topology exchange over logical links (page 107).

Database synchronization via neighbor protocol

When the first link to a peer reaches the two-way inside state of the Hello protocol, the peer nodes start the neighbor synchronization protocol. The first step of the neighbor protocol is to determine the master/slave relationship. This relationship is similar to the user/network relationship, as the master will determine the vpi.vci assignment during call setup requests. After this relationship has been established, the topology databases of the nodes begin to be exchanged until they reach full state (both nodes have the same topology view of the network).

Peer group elections

In accordance with the ATM forum PNNI specification, Nortel Multiservice Switch nodes have the ability to participate at ten levels of hierarchy. This includes being a lowest level node at one level of hierarchy and an LGN (PGL) at nine other levels.

A peer group must elect one of its members to be the PGL in order to be represented at the next higher level in the hierarchy. Also, the node's leadership priority can be configured. This allows the network operator to decide which nodes are to be considered as possible leaders for each peer group. The nodes in each peer group advertise their leadership priorities and also, their view of which node currently has the highest priority as part of their nodal information. If two nodes have the same leadership priority, the node having the highest node ID is elected. When two-thirds of the nodes agree which node has the highest priority, that node is declared the PGL. As PGL, it is responsible for creating a logical group node (LGN) at the higher level peer group to concisely represent the peer group as a single node. The PGL serves as the interface between the peer group and the LGN.

When a PGL creates a LGN, it must give the LGN an ATM end system address and a node ID that are unique within the PNNI routing domain and the physical switching system. This allows correct routing across the network and permits switched virtual connections (SVCs) to terminate at the LGN. The default setting for the ATM end system address is defined in the following order:

- 13 octet node prefix
- Shelf media access control (MAC) address octet
- selector octet

The selector octet is encoded with the hierarchy level, thus making it unique while at the same time, maintaining consistency.

The default setting for the node ID is defined in the following order:

- octet containing the level

- child peer group ID
- Shelf MAC address octet

Logical group node representation

The logical group node (LGN) aggregates the link, node and reachability information provided by the peer group leader (PGL) of the peer group it represents. The LGN uses this aggregated information to concisely describe the peer group it represents to the other LGNs in its own peer group.

Link aggregation refers to the grouping of several links to make these appear as a single link. Link aggregation only occurs at the higher levels and all links are visible within the lowest level peer groups. It is a means of summarizing the amount of link information passed around the PNNI domain. The link information describes the resources associated with ports and horizontal links such as bandwidth and ATM quality of service parameters.

The table Derivation of link resources (page 103) shows the link state attributes and metrics and how they are combined during aggregation. The information can be aggregated by taking the best case, taking the worst case, or taking an average. Nortel Multiservice Switch supports automatic link aggregation as described in the ATM Forum's PNNI Specification Version 1.0.

You can configure an aggregation token to cause links to be aggregated, especially where a link has a specific or unique characteristic that you want to make visible at the higher levels. In this case, configuring an aggregation token at both ends of the link will extricate the link from automatic link aggregation (default aggregation token value of zero). The aggregation token in combination with the node ID and port ID are used to bind the links at the higher levels. All links having the same aggregation token value between a pair of nodes will be aggregated. Configuring only one side of the link will caused the derived aggregation token to assume this value. If both sides are configured, they must be set to the same value. Configuring both ends with differing non-zero aggregation tokens will have the adverse impact of creating a derived aggregation token of zero.

Derivation of link resources

Link attribute or metric	Derived value
cell delay variation (cdv)	average value from contained links is advertised
maximum cell transfer delay (maxCtd)	average cell transfer delay is advertised
administrative weight (aw)	average value of all the contained links is used
cell loss ratio (clr)	average cell loss ratio is used

(1 of 2)

Derivation of link resources (continued)

Link attribute or metric	Derived value
maximum cell rate (mcr)	largest maximum cell rate of all contained links
available cell rate (avCr)	largest available cell rate of the contained links
(2 of 2)	

Routing calculations do not use all the metrics and attributes but select subsets of these. Subset combinations are decided by service category

Node aggregation refers to the LGN's summary view of the contained peer group. This summary view includes the information associated with link aggregation. From a routing perspective, the LGN is considered to be a single point without any associated topology metrics or attributes (values aggregated to zero).

Reachability aggregation refers to the LGN's summary view of the interior and exterior reachable addresses advertised by the nodes in the peer group that it represents. The PGL collects this reachability information and sends it upwards to the LGN, where it is subjected to further summarization and suppression. The PNNI advertisement scope associated with a reachable address specifies the hierarchy level to which the address will be advertised.

By default, the LGN has one summary address. This is the peer group ID of the peer group that it represents. The scope of an active summary address is computed to be the highest scope among all the addresses that LGN is summarizing. You can configure an LGN to summarize the reachable addresses advertised by the nodes in the peer group it represents.

You can configure an LGN to suppress the advertisement of reachable addresses (suppressed summary addresses) in the LGN's peer group, regardless of scope.

The summary or suppression of reachable addresses applies independently to internal and external reachable addresses.

LGN can advertise both summary and foreign (addresses that do not match any summary address). If a PGL sends its LGN a foreign address whose PNNI advertisement scope is higher than or equal to the level of the LGN, the LGN will advertise this foreign address. The LGN will advertise a summary address if the highest scope among all the summarized addresses is higher than or equal to the level of the LGN.

For more information, see PNNI addressing (page 48).

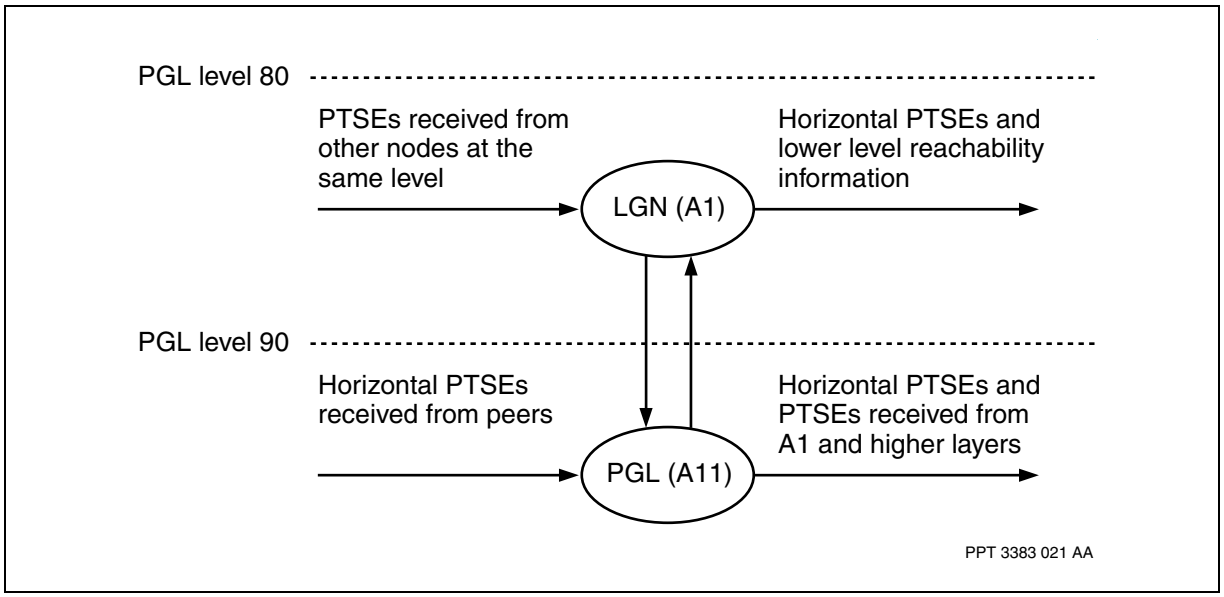
Communicating information between the PGL and LGN

In a hierarchical PNNI network, information is communicated between different levels through the PGL and LGN as illustrated in the figure Relationships between the PGL and LGN (page 105).

The downward flow of information from the LGN to the PGL can include

- nodal information
- addressing information
- link information

Relationships between the PGL and LGN



The upward flow of information from the PGL to the LGN can include

- uplink PTSEs
- addressing information
- LGN summaries
- suppressed summary addresses

The PGL is responsible for passing information describing the peer group to the LGN. Also, it is responsible for distributing the information passed down the hierarchy from the LGN to the nodes in its peer group. The LGN is responsible for distributing information from the PGL to its peers and for passing information received from its peers to the PGL.

Topology exchange between lowest level nodes

In a hierarchical PNNI network, the Hello protocol runs between the lowest level nodes (physical nodes) to discover and identify adjacent nodes. The Hello protocol also monitors any links between the nodes by exchanging peer group IDs, node IDs, and port IDs. One instance of the Hello protocol operates over each physical link or virtual path connection (VPC) through the routing control channel (RCC).

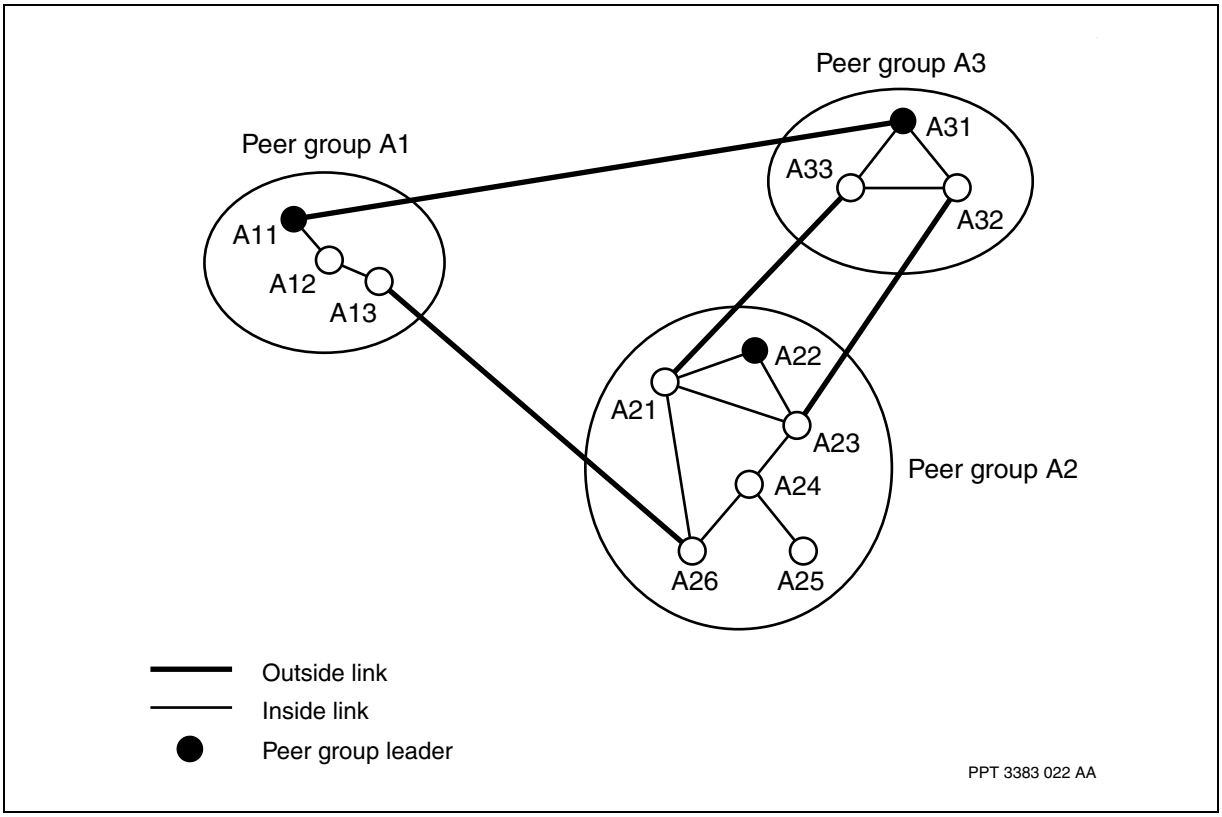
When nodes exchange the same peer group ID, the link is recognized as an inside link, meaning that it is inside the peer group. Once these adjacencies have been discovered, a Peer Database Exchange protocol runs over the RCCs on horizontal links between pairs of lowest level nodes in order to synchronize their topology databases. If different peer group IDs are exchanged, the link is recognized as an outside link, meaning that the link connects two peer groups, and that the two nodes are border nodes.

For the outside link case, the following states of the lowest level Hello protocol are supported:

- one-way outside link
- two-way outside link
- common outside link

When the common outside link is reached, this means that full bidirectional communication between the nodes has been established. The link then eligible for use in routing decisions and is flooded via the Neighbor protocol throughout the network. See the figure Relationships between link types and peer groups (page 107).

Relationships between link types and peer groups



Topology exchange over logical links

Neighboring LGNs establish routing control channels (RCC) between themselves to communicate topology and logical link information. Also, a SVCC RCC can be established between a LGN and a lowest level node (physical node) in the situation where both belong to the same peer group, and an uplink exists from the lowest level node to the LGN. Once a lowest level node or LGN receives uplink information identifying an upnode in its peer group, it creates a logical link having an associated logical port ID, and initiates procedures to establish a SVCC RCC to the upnode.

SVCC RCC calls are only required to be established in the infrequent event when a pair of LGNs discover each other. For SVCC RCCs, the on-demand routing computation will be used to determine the best available routing path. For more information, see On-demand route computation (page 111). For Nortel Multiservice Switch nodes, the preferred SVCC RCC service category is rtVbr. If a rtVbr route is not available, the call setup will be tried in the following order: CBR, then nrtVbr, then UBR.

When a SVCC RCC is being established between two LGNs, the LGN with the numerically larger node ID will initiate a SVC towards that with the numerically smaller node ID. To ensure that the correct node will be reached when the

destination node's address is being advertised by multiple nodes in the network, the destination node's node ID is used to determine the DTL for the SVC. The traffic parameters for the SVC are not configurable.

When the destination node receives the SETUP PDU, it responds with a CONNECT PDU even if it has not received an uplink PTSE identifying the source node as a neighbor. In this case, the destination node will not send Hello packets to the potential neighbor until it receives an uplink PTSE identifying the source node as a neighbor.

For Multiservice Switch nodes, you can configure a called timer and a calling timer to monitor the setup of the SVC. Upon expiry of these timers, the SVC is released.

In addition to the regular Hello protocol for the lowest level horizontal links, each LGN runs the Hello protocol over each logical link to monitor the state of the horizontal link between the two nodes. The Hello protocol on a SVCC RCC is similar to the Hello protocol between two lowest level nodes except that it deals only with states related to inside links.

The LGN Horizontal Link Hello protocol and the LGN Peer Database Exchange protocol run in parallel over the SVCC RCC. Unlike the Hello protocol which runs on the Multiservice Switch node's function processors, the LGN Horizontal Link Hello protocol runs on the control processor.

The LGN Horizontal Link Hello protocol establishes, synchronizes, and maintains the link relationship between a pair of adjacent LGNs connected by a horizontal link. Adjacent LGNs are adjacent at the logical peer group level and thus, may be physically and topologically remote from each other. Horizontal logical links represent multiple physical links. The addition or removal of physical links can happen through configuration changes or failure. The LGN Horizontal Link Hello protocol will only consider a horizontal link to be down when the last physical link associated with it becomes unavailable.

The LGN Peer Database Exchange protocol is identical to the lowest level Neighbor protocol. Each LGN uses the LGN Peer Database Exchange protocol to send and receive topology information from its peers.

Connection admission control

During route determination for an incoming call setup, the PNNI node uses generic CAC (GCAC). GCAC calculates the expected CAC behavior of another node, given the additive link metrics for that node and the requested QoS of the connection request. For more information, see the following.

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*

- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

Optimization criteria

For each node, you can configure one of the following optimization criteria for each service category:

- administrative weight (AW)
- maximum cell transfer (maxCTD)
- end-to-end cell delay variation (CDV)

Administrative weight is a metric that you assign to a link for each service class. Administrative weight represents the operational cost of the link for a given service category. For example, cost can indicate the cost for a leased line.

Links

In the context of the PNNI routing algorithm, we define the following:

- parallel links - two links connecting the same pair of nodes
- equal links - two links having the same value for the optimization metric

The PNNI routing algorithm omits links that have an optimization metric that is equal to zero. For example, if the optimization metric is administrative weight and a link has an administrative weight of zero, then the algorithm omits that link from the route calculation.

Multiservice Switch PNNI routing scheme

The Nortel Multiservice Switch PNNI routing scheme contains several features that allow

- more distributed utilization of network resources through load balancing. See Routing using path load balancing methods (page 113) and Routing using the link load balancing method (page 124).
- achievement of high call setup rates through the use of a route cache. See PNNI route cache (page 125).
- determination of the reachability of any PNNI node in the network. See Reachability of PNNI nodes (page 130).
- selection of the most effective routing path for calls to multi-homed addresses. See PNNI routing to multi-homed addresses (page 132).

- manual configuration of a designated transit list (DTL) stack containing one DTL information element (IE) for SPVC and SPVP calls. See Specified paths in flat and hierarchical PNNI (page 133).

For more information on the PNNI routing scheme, see the following sections:

- Routing techniques (page 110)
- Routing scheme steps (page 110)

Routing techniques

The Nortel Multiservice Switch PNNI routing scheme incorporates two routing techniques:

- routing using the PNNI route cache. See PNNI route cache (page 125).
- performing an on-demand route computation. See On-demand route computation (page 111).

Using these two routing techniques allows the PNNI routing scheme to accurately satisfy call setup requests having different QoS requirements while also achieving a high call setup rate. If the PNNI route cache is configured, Multiservice Switch nodes search it for an acceptable routing path that satisfies the call's QoS requirements. If the PNNI route cache does not contain any routing paths satisfying the call's QoS requirements, the PNNI routing scheme then performs an on-demand route computation to determine the routing path.

Routing scheme steps

The following procedure summarizes how the routing scheme works when routing to a non-local address:

1 Find the routing address.

The PNNI routing scheme searches for the advertised address that has the longest match against the destination address. The advertised address must satisfy the scope restrictions of the call setup.

2 Determine the set of destination routing nodes.

The PNNI routing scheme determines the set of PNNI nodes advertising the routing address. This set of destination routing nodes must satisfy the scope restrictions of the call setup.

3 Calculate a set of acceptable routing paths.

The PNNI routing scheme first tries to find the paths in the route cache. The paths in the route cache must satisfy the call's QoS requirements for all destination routing nodes.

If the route cache contains paths to all destination routing nodes, the PNNI routing scheme determines a set of acceptable routing paths from these paths, taking into consideration, the load balancing variance factors.

If the route cache does not contain paths to all destination routing nodes, the PNNI routing scheme then performs an on-demand route computation. This computation determines all the paths to the destination nodes that satisfy the multi-path variance and path diversity criteria.

4 Determine the routing path.

Based on the load balancing method selected for the call's service category, the PNNI routing scheme chooses a path from the set of acceptable routing paths calculated in step 3.

On-demand route computation

The Nortel Multiservice Switch PNNI on-demand computation scheme uses a Dijkstra-based algorithm to compute paths that satisfy multi-path variance and path diversity requirements. Once one or more paths satisfying these two requirements are found, the load balancing algorithm is used to determine the routing path.

The goal of the on-demand computation scheme is to compute multiple alternate paths to the destination nodes which are

- within the acceptable acceptance variance interval, and
- are diverse paths (paths that have the minimum of links in common).

To determine the acceptable routing paths, the on-demand route computation scheme uses a Dijkstra-based algorithm. This algorithm selects only the best optimized path to any intermediate node. If there are multiple such paths, then the algorithm selects one randomly.

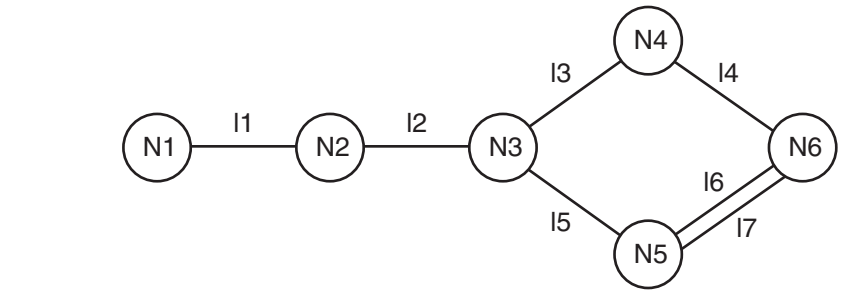
After determining the paths that satisfy the multi-path variance requirements, the on-demand route computation scheme selects only the paths that also meet the path diversity requirements. Specifically, these paths must use less than the amount of the path diversity percentage. All paths are compared to the optimal path. If there are multiple optimal paths, the on-demand route computation scheme arbitrarily selects one as the reference path for diversity. If this calculation yields a value of *pathDiversity* or greater, the path in question is considered acceptable from the path diversity point of view.

To compare path diversity, the diversity degree of path A relative to the optimal path is computed as follows:

Diversity of path A =
1 - (Number of common links)/(Number of links in the optimal path)

The figure An example of routing where all links have an equal cost (page 112) displays a network in which all the links have an equal cost.

An example of routing where all links have an equal cost



I1 to I7 = Links 1 to 7
N1 to N6 = Nodes 1 to 6

PPT 3079 001 AA

In this network, considering all links have an equal cost, routing could choose three possible paths from node one (N1) to node six (N6):

- Path 1= I1, I2, I3, and I4
- Path 2= I1, I2, I5, and I6
- Path 3= I1, I2, I5, and I7

The path diversity of path 2 relative to path 1, D (path2, path1) is as follows:

D (path2, path1)= 1-2/4= 0.5

The path diversity of path 3 relative to path 1, D (path3, path1) is as follows:

D (path3, path1)= 1-2/4= 0.5

The path diversity of path 3 relative to path 2, D (path3, path2) is as follows:

D (path3, path2)= 1-3/4= 0.25

If we assume that the acceptable path diversity is 0.4, the path diversity for both path 2 relative to path 1 and path 3 relative to path 1 is acceptable because 0.5 is greater than 0.4. However, the path diversity of path 3 relative to path 2 is unacceptable because 0.25 is less than 0.4.

For information on provisioning the *pathDiversity* component, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

If the Multiservice Switch node's PNNI on-demand computation scheme finds that there are no paths that satisfy both the multi-path variance and the path diversity requirements, it changes the optimization metric for the call setup request according to the table Optimization metrics for the PNNI on-demand computation (page 113).

Optimization metrics for the PNNI on-demand computation

Initial optimization metric	New optimization metric for re-computation
maximum cell transfer delay (maxCtd)	cell delay variation (cdv)
cell delay variation (cdv)	maximum cell transfer delay (maxCtd)
administrative weight (aw)	maximum cell transfer delay (maxCtd)

The Multiservice Switch PNNI on-demand computation scheme then uses the Dijkstra-based algorithm until all possible optimization metrics have been tried or until a path is successfully computed.

For more information on multi-path variance, see Multi-path variance (page 114).

Routing using path load balancing methods

Path load balancing occurs between multiple acceptable routing paths from the source node to the destination nodes in a Nortel Multiservice Switch PNNI network. It is based on the concept of multi-path variance and provides a better utilization of the network bandwidth.

For more information on the path load balancing method, see the following sections:

- Path load balancing process (page 114)
- Multi-path variance (page 114)
- Path load balancing options (page 115)
- Load balancing in hierarchical PNNI networks (page 121)

- Load balancing on available cell rate in hierarchical PNNI networks (page 122)

Path load balancing process

In Nortel Multiservice Switch PNNI routing, path load balancing is a two-step process:

1 The node obtains a set of acceptable routing paths via the on-demand route computation or the route cache.

2 The path load balancing scheme then selects a routing path from the set of acceptable routing paths, based on the load balancing method specified by the network operator.

Multi-path variance

The multi-path variance is specified on the basis of the ATM service category and the optimization metric supported by the service category. The characterization of the set of acceptable routing paths is based on configurable variance attributes, *minVariance* and *slopeVariance*, both under the *ARtg Pnni LoadBalancing* component. These attributes are used to calculate the maximum optimization metric value of all acceptable paths:

MaxOptMetric = OptMetric(bestPath) + acceptableVariance

Acceptable routing paths are all paths that satisfy a call's traffic requirement and have an optimization metric smaller than the optimization metric of the optimal path plus an acceptable variance. The acceptable variance is computed as follows:

*acceptableVariance = minVariance + OptMetric(bestPath)*slopeVariance/100*

An optimal routing path is a path which minimizes the optimization criteria and also meets a call's QoS requirements (see Multiservice Switch PNNI routing scheme (page 109)).

An acceptable path is a routing path for which the optimization metric is within the defined variance interval and which meets the call's QoS requirements. Allowing optimal and sub-optimal paths to be considered as acceptable paths increases the number of paths that can be used for routing calls. This results in a more balanced utilization of the network's resources and a reduction of network congestion.

This allows operator control of the feature based on the service category and the constraints imposed by that service category.

See NN10600-060 *Nortel Multiservice Switch 7400/15000/20000 Component Reference* for component and attribute information.

Path load balancing options

Among the acceptable routing paths, the Nortel Multiservice Switch PNNI routing algorithm selects a path based on the load balancing options chosen by the network operator. Selection of a routing path can be

- on the basis of uniform distribution
- on the basis of the widest path
- in proportion to the available cell rate
- in proportion to the optimization metric
- in proportion to both the available cell rate and the optimization metric

The uniform distribution method randomly selects an acceptable path.

The widest path is the path with the highest minimum available cell rate. The widest path method allows a better distribution of the load in the network by selecting the paths that have more available bandwidth. Therefore, this method avoids potential network congestion when the load could be shared among other resources. This method introduces an efficient and effective way to process load balancing at the access nodes and in the core of the network.

By definition, to find the widest sub-path/path, the algorithm compares the minimum AvCr of each path and selects the path with the highest minimum AvCr. By definition, the minimum AvCr of a path is the minimum AvCr of the segment of the path that is not common to the path against which it is compared. The following procedure is the algorithm that is used to select or prune a sub-path/path:

1 Compare the optimization metrics of the paths.

   a. Choose the less costly path.

   b. If the metrics are the same, then proceed to step 2.

2 Compare the minimum available cell rates of the two paths.

   a. Choose the greater of the two.

   b. If the constraints are equal, then proceed to step 3.

3 Use the Mark and Sweep algorithm to determine which segment is preferred. The Mark and Sweep Algorithm is used when there are multiple optimal sub-paths to an intermediate node. It selects the best path among acceptable paths. For more information about the Mark and Sweep Algorithm, see Mark and Sweep algorithm (page 119).

   a. If the Mark and Sweep algorithm returns a path, choose it.

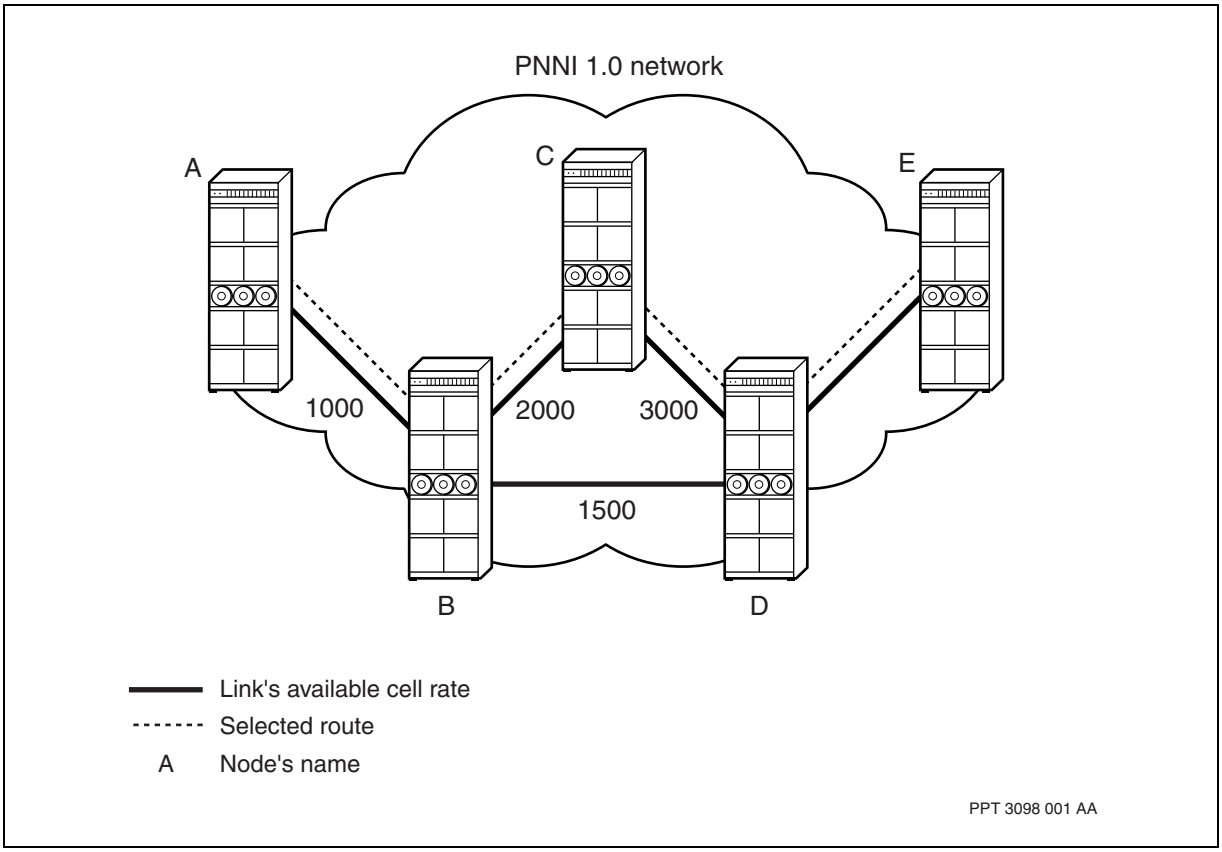
b. If the result is indeterminate, which occurs if the two segments are equally constrained, then proceed to step 4.

4 Choose randomly. The randomization function ensures that all paths to a common node have an equal chance of being selected regardless of the order in which they were expanded.

Attention: At the destination node, if variance is enabled, paths within the variance interval of the best path are considered to have equal cost for the check in Step 1.

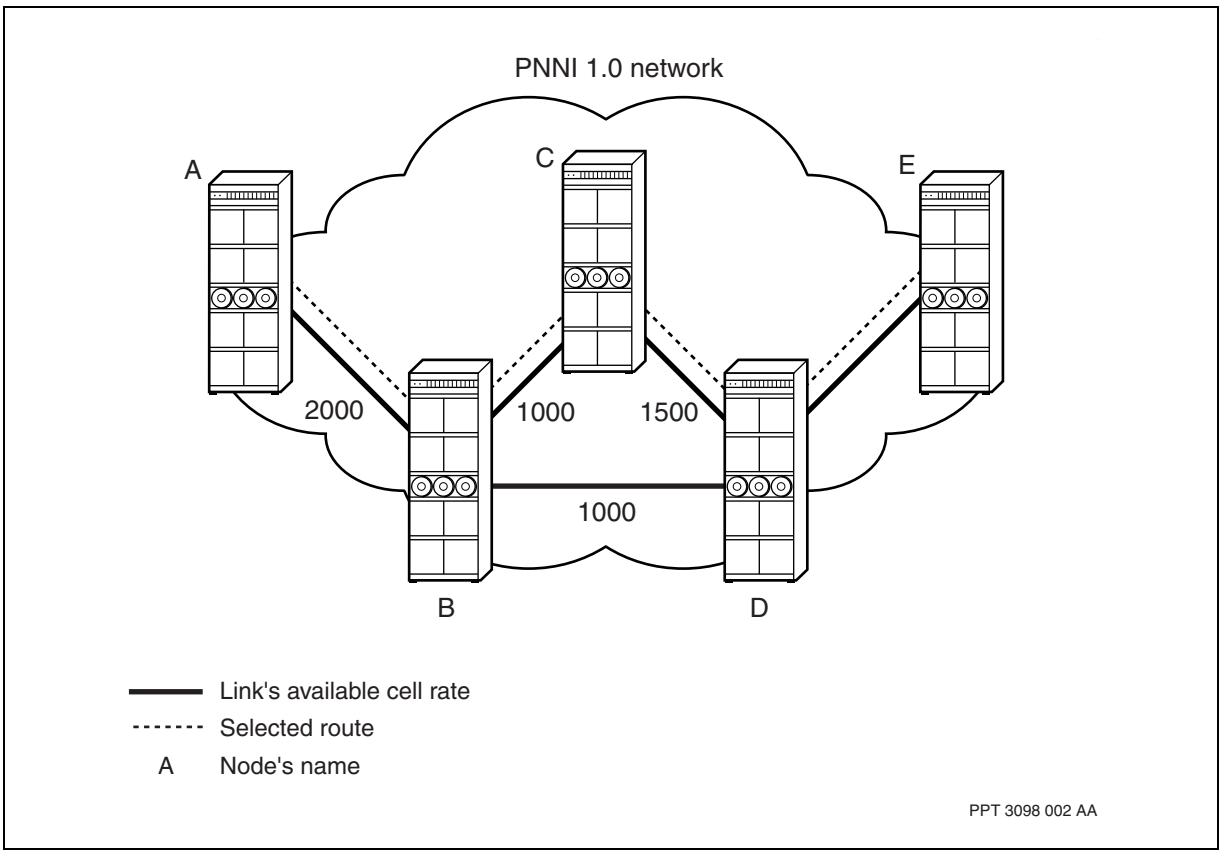
The following example demonstrates that using only the minimum available cell rate of a path is not sufficient in selecting the path with the highest bandwidth. The figure An example of different AvCr on distinct path segments (page 117) shows a network where the minimum available cell rate is on the common link of the two acceptable paths. Assuming that sub-path1{ABCD} and sub-path2{ABD} are both optimal sub-paths, then the widest sub-path should be selected. However, in this example, the minimum available cell rate of both paths is the same (1000) and the limiting bandwidth comes from the common link {AB}. To perform a better selection, the algorithm compares the minimum available cell rate of the distinct segments of the paths which are for sub-path1 {BCD} (2000) and for sub-path2 {BD} (1500). Sub-path1 is selected because this sub-path has a higher minimum available cell rate on its distinct segment than sub-path2.

An example of different AvCr on distinct path segments



In the previous example, without the notion of distinct segment, both sub-paths are considered equal in bandwidth. Therefore, the selection is done randomly and the sub-path with less bandwidth might be selected. The next example shows a case where the selection of the sub-path is done randomly. The figure An example of same AvCr on distinct path segments (page 118) shows a network where the equal minimum AvCr is on the distinct segments of the two optimal sub-paths. Assuming that sub-path1 {ABCD} and sub-path2 {ABD} are both optimal sub-paths, then the widest sub-path must be selected. In this example, both sub-path1 and sub-path2 have a minimum AvCr of 1000. However, this minimum AvCr comes from their respective distinct segments. In such a situation, sub-path1 and sub-path2 are considered to have the same amount of bandwidth and therefore the best sub-path is randomly selected.

An example of same AvCr on distinct path segments



The proportional to the available cell rate method randomly selects an optimal sub-path with regards to the optimization metrics at the intermediate nodes. At the destination node, this method proportionally selects the optimal path based on the available cell rate.

The proportional to the optimization metric method randomly selects an optimal sub-path with regards to the optimization metrics at the intermediate nodes. At the destination node, this method proportionally selects the optimal path based on the available cell rate.

The proportional to the available cell rate and the optimization metric method randomly selects an optimal sub-path with regards to the optimization metrics at the intermediate nodes. At the destination node, this method proportionally selects the optimal path based on the available cell rate and optimization metric.

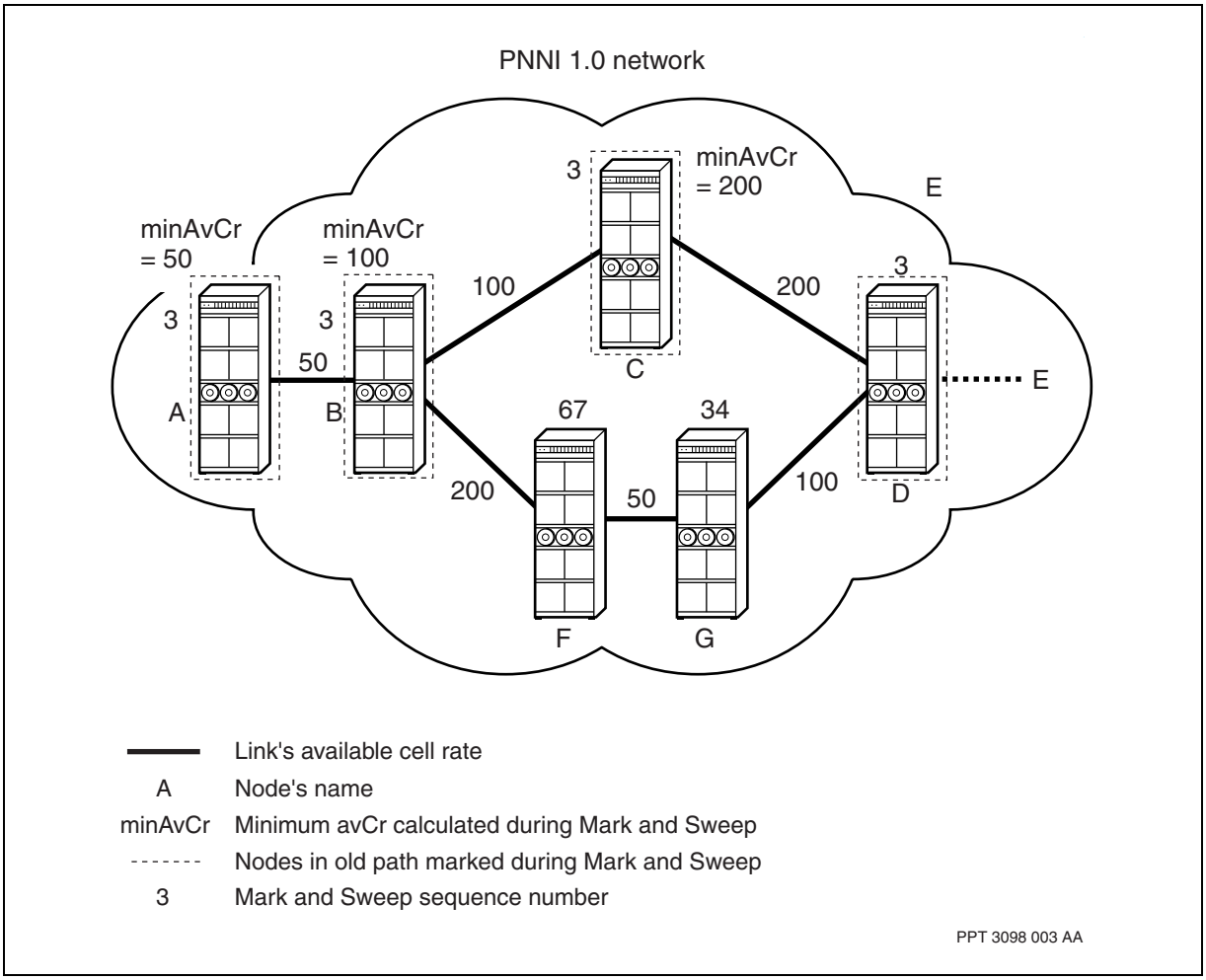
Mark and Sweep algorithm

The Mark and Sweep algorithm is used to compare the minimum available cell rate of two sub-paths/paths in their distinct segments. This algorithm works by marking all the nodes of one path and then sweeping back along the second path until the common node is encountered. The common node can be a common intermediate node or the source node.

In the figure An example of the Mark and Sweep algorithm at an intermediate node (page 120), the first optimal sub-path computed to node D is Path1 {ABCD}. Then, a second optimal sub-path is found, Path2 {ABFGD}. In this example, both sub-paths have the same path minimum AvCr (50). The Mark and Sweep procedure must then be applied.

First, the first sub-path (Path1) is marked towards the source node. Therefore, the nodes C, B and A are marked. Then the second sub-path (Path2) is traversed towards the source node until either a marked node or the source node is encountered. In the present case, the common node B is encountered. During the process of marking Path1 and sweeping Path2, the distinct segment minimum AvCrs are calculated. For Path1, the minimum AvCr for the distinct segment is at B=100. For Path2, the minimum AvCr is kept during the sweeping progress. Once B is recognized as the divergent node, the minimum AvCr for Path1 stored in node B (100) is compared with the minimum AvCr found while sweeping Path2(50). The highest minimum AvCr determines what path is to be selected. In the example, Path1 is selected because its minimum AvCr on its distinct segment is 100 and the minimum AvCr of the distinct segment of Path2 is 50.

An example of the Mark and Sweep algorithm at an intermediate node



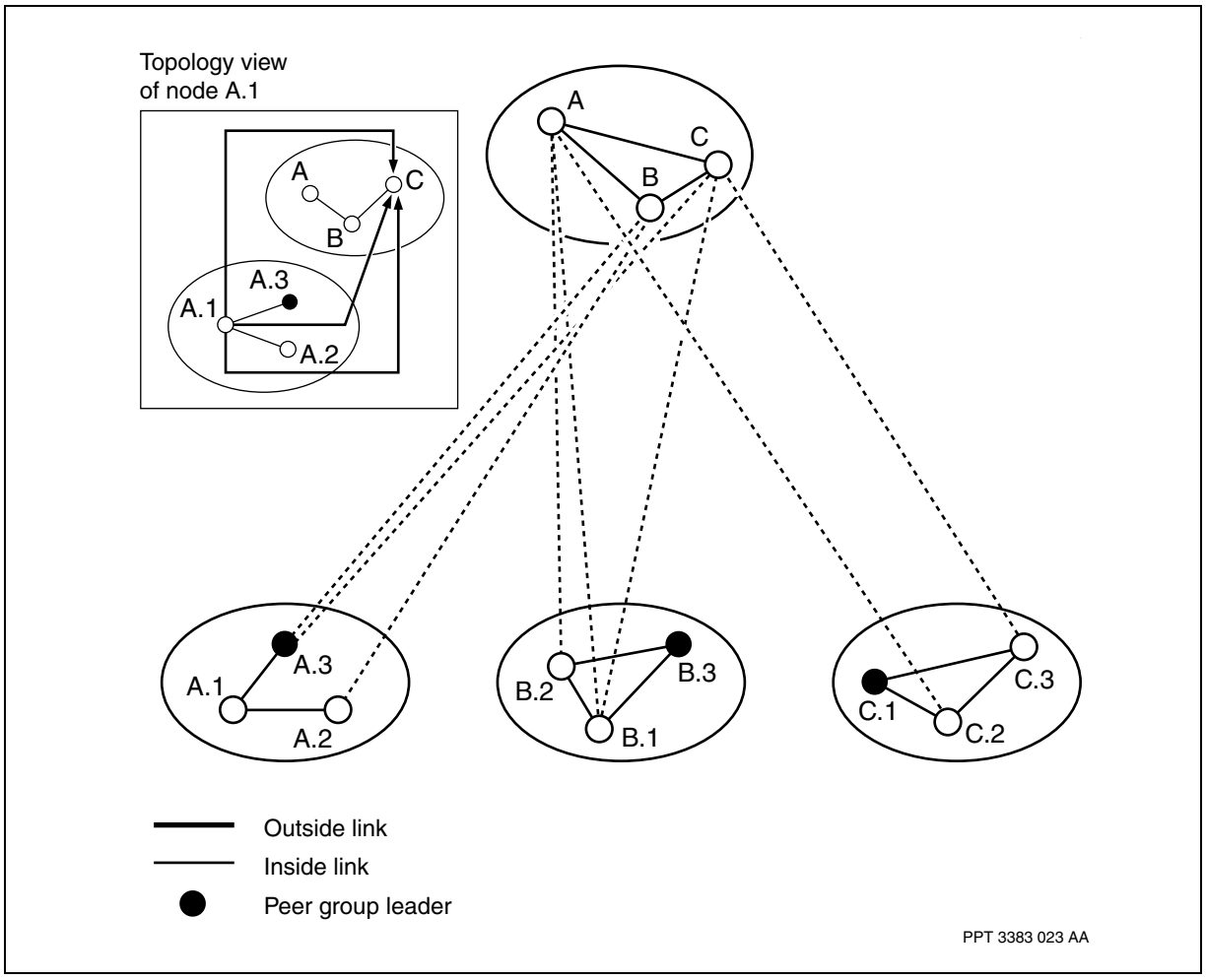
If the source node is encountered instead of a common node, this means that both sub-paths/ paths are completely distinct. Therefore, the best sub-path/ path is selected randomly. This is because, in that case, the distinct segments are the entire paths and the algorithm already verified that the sub-path/path minimum AvCr is the same. The same process applies for the Mark and Sweep at the destination node. This algorithm is used every time there are multiple optimal sub-paths to an intermediate node. It is also used at the end of the Dijkstra computation to select the best path among acceptable paths. At an intermediate node, the Mark and Sweep algorithm is used N-1 times, where N is the number of optimal sub-paths to a node that have equal path minimum available cell rates. For the destination node, the Mark and Sweep algorithm is used M-1 times, where M is the number of acceptable paths with the same path minimum available cell rate. The processing of the Mark and Sweep is all done on the source node. The reference to the intermediate and destination nodes is simply a topology reference inside the route computation.

Load balancing in hierarchical PNNI networks

In hierarchical PNNI networks, path load balancing occurs at the Nortel Multiservice Switch PNNI node acting as the source node of a connection. That is, the Multiservice Switch load balancing scheme performs load balancing at the network level, within the PNNI source node's peer group and between its peer groups.

The figure, An example of load balancing in a multi-level PNNI network (page 121), illustrates the case where a Multiservice Switch node, node A.1, is the originator of the call.

An example of load balancing in a multi-level PNNI network



The topology view of the source node A.1 shows that there are three possible routing paths:

- A.1, A.3, C
- A.1, A.2, B, C

- A.1, A.3, B, C

If the three routing paths satisfy the call's QoS requirements and their optimization metrics are within the load balancing variance interval, then the source node considers all these paths as acceptable routing paths. The PNNI routing scheme then chooses one routing path according to the specified load balancing method.

Load balancing on available cell rate in hierarchical PNNI networks

The link load balancing method (maxAvCr) in a PNNI network requires some understanding of how significant changes are propagated throughout the network. A significant change is an event that occurs in which the information about that event is immediately flooded throughout the PNNI network. This event can be things like a change in administrative weight or as it specifically relates to this link load balancing, a change in available cell rate on a given link. This information must be propagated by PNNI to ensure accurate topology information for routing purposes.

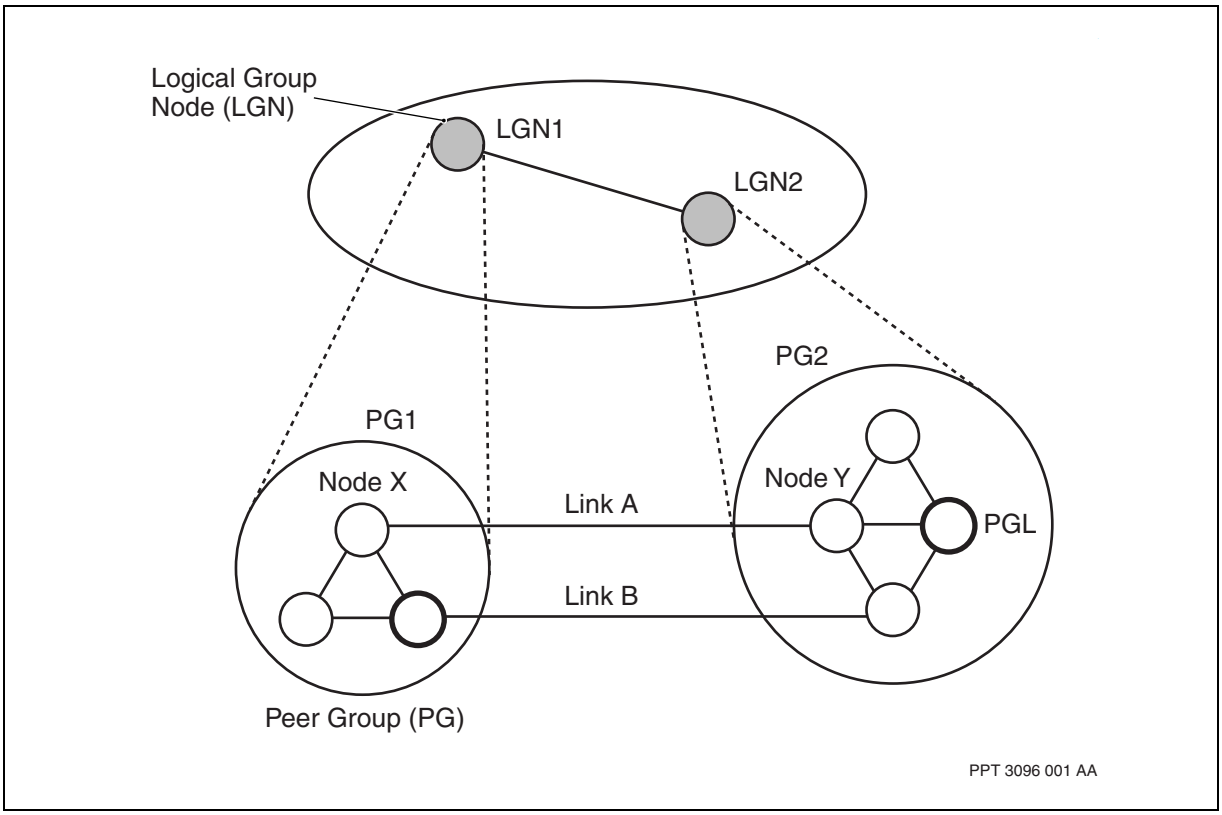
In PNNI, each node that sources its own PNNI topology state elements (PTSEs) will refresh them every 30 minutes by default. Thus, if there is not a significant change to the PTSEs of that node, then all the other nodes in the peer group (PG) will not age this PTSE until one full hour. Therefore, if no threshold is crossed it will take the node sourcing its own PTSEs half an hour before any updates will be advertised on the other nodes in the peer group.

The load balancing on available cell rate in hierarchical PNNI networks feature enhances the widest path load balancing method (maxAvCr). For more information on the widest path load balancing method, see Path load balancing options (page 115).

This feature extends the nodal threshold parameters (avCrMt and avCrPm) to a link level. Therefore, with this granularity it is recommended in HPNNI that the nodal threshold parameters of the logical group node (LGN) for the higher level be set identically to the minimum of the link threshold parameters of all the outside links leaving that peer group on that level. When a link threshold is crossed, (significant change) the new link information is first propagated to the PGL, which in turn uses the nodal threshold values at that level to determine if further propagation is required. Therefore, changes to the link threshold should correspond to the nodal threshold values of the LGN. Be aware that the higher level node (LGN) may be a different physical node than the node on which the link values are set. Therefore, changes to a link threshold may need to accompany changes to nodal threshold changes at another node in the peer group.

In the following figure, An example of load balancing on available cell rate for HPNNI networks (page 123), there are two outside links to the same neighboring LGN and that have different link thresholds defined on each link. On Link A, the avCrMt and avCrPm are set to 1% respectively and on Link B the values are the default values: avCrMt 3% and avCrPm 50%.

An example of load balancing on available cell rate for HPNNI networks



Assuming that both links are OC3 and the available cell rate per link is 353 257 cells per second (cps). It would only take a call traversing Link A with bandwidth requirement of 3600 cps or greater to trigger a significant change since 3600 cps is greater than 1%. If a call is setup with more than 3600 cps, then a significant change will occur on Link A and will be broadcast within the lowest level peer group. However, since the PGL will use the maximum value (50%) for the horizontal link at the higher level (remember it uses the maximum of the two links), the LGN will not issue an advertisement based on the significant change that occurred at the lowest level.

Therefore, it is emphasized that the nodal threshold values for the LGNs and for Node X and Node Y are provisioned identically to the link threshold values on all outside links. This will ensure that LGN will advertise the significant change at the higher level of hierarchy.

Routing using the link load balancing method

In the Nortel Multiservice Switch PNNI network, link load balancing is performed at a transit PNNI node when the egress port identifier is not specified in the corresponding designated transit list (DTL) element of the path. In this situation, the Multiservice Switch node has to determine the link to the next PNNI node specified in the path. If there are multiple links between the Multiservice Switch node and next PNNI node specified in the path, link load balancing distributes the traffic load between the acceptable routing links.

For more information on the link load balancing method, see the following sections:

- Acceptable routing links (page 124)
- Link load balancing options (page 124)
- Selection of the link load balancing options (page 125)

Acceptable routing links

Acceptable routing links are all the links between the Nortel Multiservice Switch node and next PNNI node specified in the path that satisfy the call's traffic requirements and also minimize the optimization metric. Unlike path load balancing, link load balancing does not consider a variance factor in determining a set of acceptable routing links.

Link load balancing options

From the set of acceptable routing links, the Nortel Multiservice Switch PNNI routing algorithm selects a routing link based on the one of three load balancing options:

- widest link
- uniform
- proportional to the available cell rate

The widest link option chooses the routing link that has the highest available cell rate. It will keep using the chosen routing link until it is no longer the widest link.

In uniform load balancing option, each acceptable routing link has an equal chance of being selected. Over time, each acceptable routing link is chosen an equal number of times.

Selecting a routing link in proportion to the available cell rate means that the chance of selecting an acceptable routing link is proportional to the available cell rate of the acceptable routing links.

Selection of the link load balancing options

The Nortel Multiservice Switch PNNI routing scheme chooses the link load balancing option on the basis of the load balancing method configured by the network operator under the *ARtg Pnni LoadBalancing* component through the *method* attribute (See NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.). The attribute values and their corresponding link load balancing options are summarized in Link load balancing options (page 125).

Link load balancing options

Value of ARtg Pnni LoadBalancing method attribute	Link load balancing option
maxAvCr	widest link
random	uniform
avCrProb	proportional to the available cell rate
optMetricProv	uniform
avCrOptMetricProv	proportional to the available cell rate

PNNI route cache

The PNNI route cache stores and maintains multiple routing paths to several destinations. It uses the stored entries in the route cache to provide routing paths for subsequent call setup requests, thus increasing the call setup rate and reducing call setup latency. If an appropriate entry in the route cache cannot be found, then an on-demand route computation is performed.

For more information on the PNNI route cache, see the following sections:

- Organization and size (page 125)
- QoS profile matching (page 127)
- Routing cache search (page 128)
- Creating and maintaining route cache entries (page 129)
- Dynamic learning scheme (page 129)
- Route replacement operation (page 129)
- Route purging operation (page 129)

Organization and size

The PNNI route cache is a collection of routes classified with respect to

- service category (CBR, rtVBR, nrtVBR, UBR (including UBR with MDCR)
- optimization criteria

- QoS profile
- destination node

Each route consists of multiple alternate routing paths to a particular destination node. Routing path attributes are collectively referred to as a QoS profile. See the table QoS profile attributes (page 126) for their applicability to the ATM service categories.

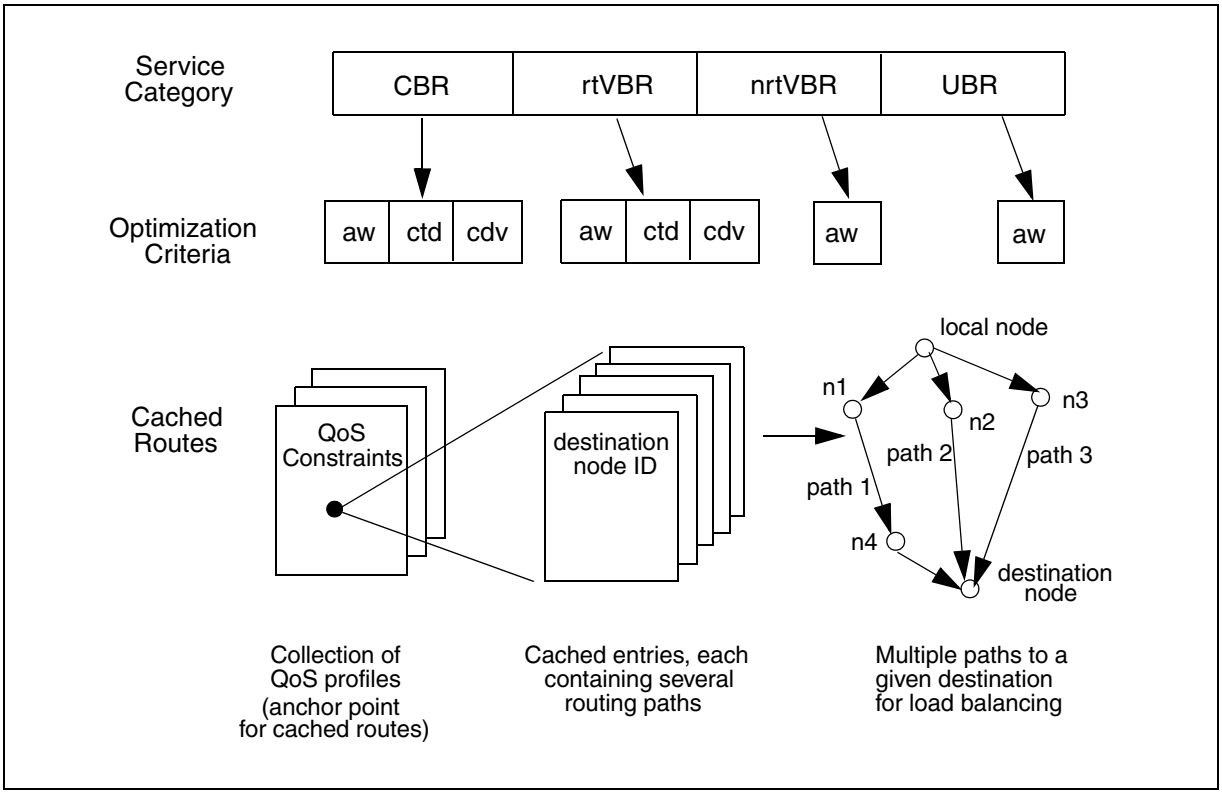
QoS profile attributes

Attributes	Applicability to ATM service categories
minFwdCR	minimum forward cell rate for CBR, rtVBR, nrtVBR, and UBR
minBwdCR	minimum backward cell rate for CBR, rtVBR, nrtVBR, and UBR
ctd	cell transfer delay for CBR and rtVBR
cdv	cell delay variation for CBR and rtVBR
clr	cell loss ratio for CBR and rtVBR

To enhance search performance, the route cache is organized hierarchically. It contains three levels as shown in the figure Structure of the PNNI route cache (page 127).

The top level contains four different pools, one for each service category supported by Nortel Multiservice Switch. The next level in the hierarchy contains the optimization metrics supported for each ATM service category. The lowest level contains the collection of cached routes corresponding to a given service category and given optimization metric. In these pools, the cached routes are classified and stored based on the QoS profiles. Furthermore, to facilitate load balancing, each cached route is stored as a collection of *Lb/<srvCat> maxPaths* alternate routing paths.

Structure of the PNNI route cache



You can set the size of the route cache by specifying

- maximum number of routes (*ARtg Pnni Cache maxNumEntries*)
- number of alternate routing paths (*ARtg Pnni LoadBalancing maxPaths*)

For information on setting the size of the route cache, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

QoS profile matching

The Nortel Multiservice Switch PNNI routing scheme uses a profile matching operation to find a suitable route in the route cache for each call setup request. If the routing scheme cannot find a route that matches the QoS profile of the call setup request, it selects a cached routing path whose QoS profile most closely matches that of the call setup request. If the profile of the cached route and that of the call setup request differs by an unacceptable amount that is internally defined, then the cached route is not considered to be an acceptable match. In such cases, Multiservice Switch will perform the on-demand route computation.

In profile matching, the PNNI routing scheme always considers a cached route that has a better QoS profile than that of the call setup request. For example, if a call setup request for a 1.4Mbps connection with a CTD requirement of 325 milliseconds is received and a cached routing path with an exact profile match is not available, the PNNI caching scheme would select a cached routing path satisfying 1.5 Mbps and a CTD of 300 milliseconds.

See the table QoS profile matching requirements (page 128) for a summary of conditions that must be satisfied for a cached routing path to be considered a matching route. Note that the Multiservice Switch PNNI routing scheme will only compare QoS profiles among routes with the same service category and the same optimization metric.

QoS profile matching requirements

Optimization metric	Requirement for QoS profile matching
minFwdCR	The call setup request's minFwdCR is less than or equal to the minFwdCR value against which the cached routing path was computed.
minBwdCR	The call setup request's minBwdCR is less than or equal to the minBwdCR value against which the cached routing path was computed.
ctd	The call setup request's CTD is greater or equal to the cached routing path's CTD
cdv	The call setup request's CDV is greater than or equal to the cached routing path's CDV
clr	The call setup request's CLR is less than or equal to the cached routing path's CLR.

Routing cache search

In the Nortel Multiservice Switch PNNI routing scheme, searching for a routing path in the routing cache is a three-step process:

1 Find the QoS profile pool.

The PNNI routing scheme selects the cache pool corresponding to the call's service category and optimization metric.

2 Find a matching QoS profile.

If the QoS profile matching operation finds a matching QoS profile in the route cache, it uses this route cache entry in the call setup. If a cached route having an appropriate QoS profile is not found in the route cache, the PNNI routing scheme performs an on-demand route computation and the cache learns this computed routing path.

3 Choose a routing path.

A cached route can consist of multiple alternate routing paths to a destination node. The PNNI routing scheme selects a routing path according to the specified load balancing method.

Creating and maintaining route cache entries

As well as through the profile matching operation, the PNNI routing scheme maximizes the call setup rate through the dynamic learning of routing paths and maintenance of the route cache so the entries are up-to-date. Maintaining the route cache involves

- knowledge of the latest view of the network topology
- dynamic learning scheme
- route replacement operation

Dynamic learning scheme

All of the routing paths, including their QoS profiles stored in the PNNI route cache, are learned as a result of call setup requests whose QoS profiles do not have acceptable matches in the route cache. That is, when the profile matching operation fails to find a suitable routing path from the route cache, the PNNI routing scheme performs an on-demand route computation. The resulting computation is used to route the call and is also learned by the route cache.

Route replacement operation

When the maximum number of QoS profiles allowed for a given service category is reached, a stored routing path needs to be removed from the route cache to make room for a newly learned routing path. The process of selecting a suitable routing path for removal is called the route replacement operation.

The route replacement operation is based on the age of the stored routing paths. Nortel Multiservice Switch selects all the routing paths forming the 5% oldest routes in the route cache for removal from the route cache.

Route purging operation

The route purging operation maintains the validity of the cached routing paths. Cache entries are valid only for a limited duration due to the dynamic nature of the PNNI network. Also, link or node failures may affect the integrity of the routing paths stored in the cache. See the table Route purging descriptions (page 130) for descriptions of the Nortel Multiservice Switch routing scheme's purging operations. For more information on rerouting and optimizing commands, see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

Route purging descriptions

Type	Description
Aging	Every profile in the route cache is subject to an aging process. When a profile reaches the maximum lifetime specified by the *ARtg Pnni Cache agingPeriod* attribute, the PNNI routing scheme removes it from the cache
Crankback purging	In the event of a crankback, the PNNI routing scheme removes all multiple alternate routing paths for the cached route containing the link or node causing the crankback. If there are alternate routing paths that do not involve the link or node causing the crankback, the PNNI routing scheme selects one to avoid an on-demand route computation for that crankback request. If *purgeEntryAtPathInvalid* is set to yes, the whole route cache entry (including all multiple alternate paths) will be purged after a path becomes invalid. If this attribute is set to no, only the path will be removed. For most networks, it is desirable to purge the whole route cache entry when one path becomes invalid since this allows for better load balancing. However, for some network scenarios, this behavior is not desirable, because it may lead to many unnecessary route cache purges at high network utilization.The default value of this attribute is yes. It should only be changed if recommended by Nortel Networks.
Significant resource changes	If the PNNI routing scheme receives a PTSE reporting significant resource changes affecting links or nodes, it removes all routing paths affected by the change. If *purgeEntryAtPathInvalid* is set to yes, the whole route cache entry (including all multiple alternate paths) will be purged after a path becomes invalid. If this attribute is set to no, only the path will be removed. For most networks, it is desirable to purge the whole route cache entry when one path becomes invalid since this allows for better load balancing. However, for some network scenarios, this behavior is not desirable, because it may lead to many unnecessary route cache purges at high network utilization.The default value of this attribute is yes. It should only be changed if recommended by Nortel Networks.
complete purging	When too many changes occur in the network topology, you should use the operational clear command to remove all the QoS profiles from the route cache.

Reachability of PNNI nodes

The Nortel Multiservice Switch PNNI routing system provides support for determining the reachability of any PNNI node from the local node. In a PNNI network, a node is considered reachable from the local PNNI node if there is

a path that connects it to the local node. During the lifetime of a PNNI network, some nodes can lose their connectivity to other nodes in the network. However, the local node's PNNI topology database contains the nodal information PTSEs for these nodes until they age out. The Multiservice Switch node's ability to determine which PNNI nodes are reachable from the local node eliminates attempts to route calls to unreachable nodes.

For more information on the reachability of PNNI nodes, see the following sections:

- Reduction in crankbacks (page 131)
- Reachability spanning tree (page 131)

Reduction in crankbacks

The availability of reachability information of the PNNI nodes in the topology avoids the generation of unnecessary crankbacks. At the terminating node, if there is no connectivity to the called party and if the terminating node's link state database indicates that the called party can be reached through another PNNI node that is not an ancestor, the ATM Forum PNNI Version 1.0 routing specification requires that the call be cranked back. The Nortel Multiservice Switch node checks the reachability of the PNNI node that advertises the called party. If the PNNI node advertising the called party is not reachable from the terminating node, it is also not reachable from the source node so the call is released instead of being cranked back.

For information on crankback, see Crankback mechanisms (page 165).

Reachability spanning tree

Once a reachability spanning tree from the local PNNI node is generated, you can display the reachability status of a PNNI node in the topology. To match the dynamic nature of the network, the Nortel Multiservice Switch node updates the reachability spanning tree when one of the following critical topology updates occurs:

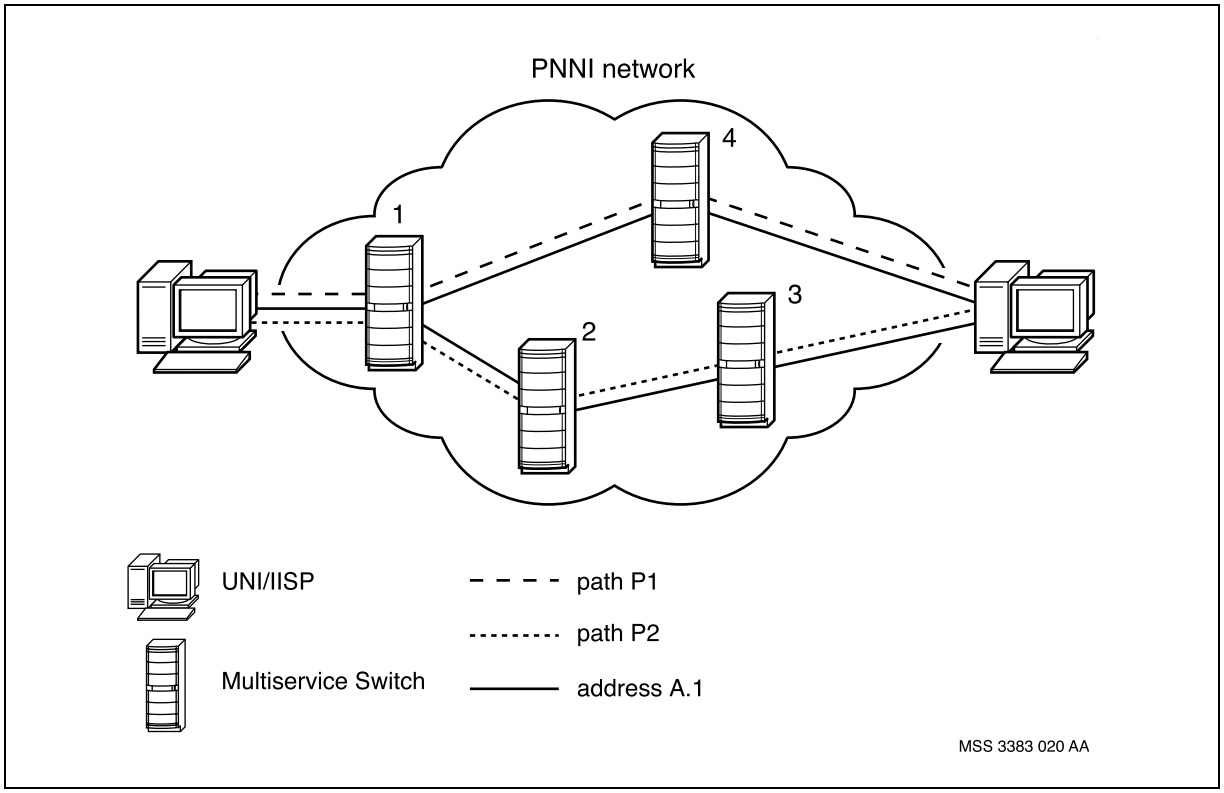
- adding a new link to the topology
- deleting a link from the topology
- adding a new node to the topology
- deleting a node from the topology.

For information on displaying reachability information for a PNNI node, see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

PNNI routing to multi-homed addresses

Multi-homed addresses are addresses that are advertised by at least two PNNI nodes. In the figure, An example of multi-homed addresses (page 132), for redundancy purposes, the same end device is connected to two Nortel Multiservice Switch nodes, 3 and 4. In such cases, if the end device registers the same address on the links connecting it to the Multiservice Switch nodes, then both nodes will advertise the same address.

An example of multi-homed addresses



The Multiservice Switch PNNI routing scheme permits the selection of the routing path from a set of acceptable paths from the source to the various destination nodes advertising the multi-homed address. If the load balancing variance factors generate acceptable routing paths to the different nodes advertising the destination address, then the connections to the destination address are distributed among the different non-PNNI links to the destination end device. Load balancing can still be observed between the links to the end device.

Load Re-Balancing on Parallel Links

Whenever a new ATM Private Network-Network Interface (PNNI) parallel link is added, or an existing PNNI parallel link is dropped and then recovered, Load Re-Balancing on Parallel Links (LRB) distributes the bandwidth equally across the parallel links. This increases the overall throughput in the ATM PNNI network, and reduces the chance of congestion.

LRB is activated by provisioning the parallel links as a group. LRB supports up to five parallel links in one group, and up to five groups per node. A link can belong to only one group.

For connections with the Equivalent Cell Rate (ECR) equal to 0, LRB re-distributes the number equally across the parallel links.

Specified paths in flat and hierarchical PNNI

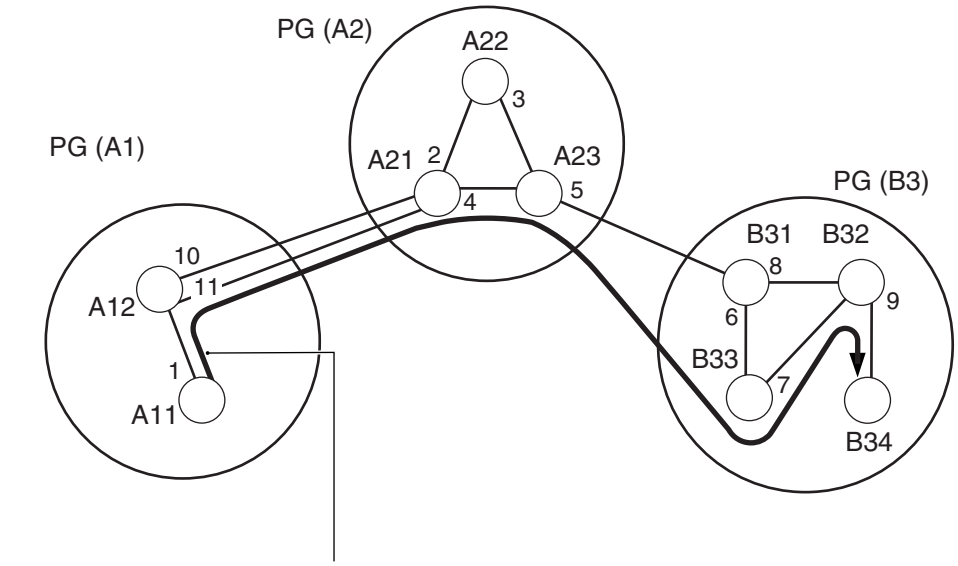
Specified paths for flat and hierarchical PNNI enable the network operator to manually configure a path with a predetermined sequence of physical nodes and port IDs for SPVP/VPT SPVP and point-to-point SPVC calls in a flat and hierarchical PNNI topologies. The SPVP and point-to-point SPVC calls are then transformed into a designated transit list (DTL) stack at call setup time.

Two specified paths can be provisioned for each SPVP and point-to-point SPVC call: the primary path and the alternate path. The primary path is mandatory and the alternate path is optional. Initially, the primary path is used to establish the point-to-point SPVC and SPVP call. If the primary path fails for any reason during call establishment, the alternate path is used. In the case where an alternate path was not provisioned or has failed, PNNI routing will attempt to find a route if the automaticFallback attribute is set to enabled.

A specified path consists of an ordered list of 2 to 40 hops. The path must include the source node and the destination node. The list of hops specifies a connected sequence of nodes and links from the source node to the destination node as is displayed in figure An example of a specified path (page 134).

A specified path connection will still be established even if a node along the specified path has the Artg Pnni configuredNode/* restricTransit component set to true at the lowest configurable level of PNNI.

An example of a specified path



A specified path must be complete from source to destination. Partially defined paths are not supported. In the figure An example of a specified path (page 134), the correct list of hops determined by the operator is: A11, A12, A21, A23, B31, B33, B32, and B34. If the operator forgets to add hop A12, the specified path connection using that particular specified path will fail because the port ID 1 of node A11 does not connect directly to A21.

If a port ID specified in a path does not match an existing port ID on the associated node, then the specified path will fail to connect. The specification of port IDs on the specified path is optional, and by default the port ID is zero. In which case, a port ID on the next node in the DTL stack (built from the specified path information) will be selected by the node. If multiple paths exist between the nodes then the existing Multiservice Switch load balancing functionality will apply across the available paths.

Since the specified path did not specify the transition to take from A12 to A21 in figure An example of a specified path (page 134), the enhanced routing system will have to select an egress port ID to reach A21, either 10 or 11.

It is recommended to set the port ID to zero for any specified path using a third party node that supports a port ID value of zero because the port IDs could potentially change whenever cards are reset. The port ID value for the last hop (destination node) in the list of hops, (that is, the destination node) is not necessary and it is recommended that the port ID be set to zero. By default when adding a hop to the list of hops, the port ID value is zero.

Specified path validation is performed at provisioning time by the semantic check to verify that all the node names in the path are present in the node name translation table. Validation also ensures that no duplicate hops are provisioned in the path and that the path does not exceed 40 hops.

To use the PNNI Path Trace to validate a specified path composed of 20 nodes or less, only the source node needs to be upgraded with the Specified Paths in Flat and Hierarchical PNNI software (PCR4.2). However, all the nodes need to have at least PCR4.1 software. To use the PNNI Path Trace to validate a specified path composed of more than 20 nodes, it is recommended to upgrade all the nodes along the path with the Specified Paths in Flat and Hierarchical PNNI software to ensure that the length of the TTL IE does not exceed the maximum length supported (increased from PCR4.1 to PCR4.2 to support 40 hops tracing). If you reach the maximum message length allowed, the node will set the Trace status to "trace has exceeded information element length limitations" or "trace has exceeded message length limitations". For more information about path trace, see PNNI path trace (page 87).

More validation can be performed by the operator after the specified path is provisioned and activated. The operator can verify the validity of a specified path by executing the Trace command on that particular path with a set of QoS requirements. For more information about the Trace command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*. For more information about crankback, see Crankback for specified paths (page 176).

The Multiservice Switch node is implementing specified path functionality using the lowest level physical node IDs.

If a loop exists in a hop list, an error will be generated by semantic check.

The following nodes must have PCR4.2 software loaded in order for specified paths to function properly:

- source node
- any exit border node
- the twentieth hop node along the specified path

A Multiservice Switch PNNI node can be restricted from being used as a transit node. When a node is transit restricted, some information is exchanged within its peer group and the node is excluded by the PNNI routing system when performing a route calculation. This information is not advertised outside its respective peer group. When a node along a specified path is restricted as a transit node by the PNNI routing system, the specified path connections will ignore the restrict transit setting and the specified path connections will be established through that node.

Specified paths are configured through the Mdtl and MdtlPath components. For more information about configuring these components, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

For more information on configuring specified paths using the Nortel Multiservice Data Manager, see NN10400-006 *Nortel Multiservice Data Manager Network Configuration*.

PNNI connection recovery and path optimization

Nortel Multiservice Switch nodes support two types of rerouting services operating in a PNNI or HPNNI network for active point-to-point SVC, SVP, soft PVC, and soft PVP connections as follows:

- Connection recovery (or hard rerouting): The rerouting of a connection segment which is released due to a network failure at one or more points on the connection segment.
- Path optimization (or soft rerouting): The rerouting of a connection segment to obtain a better path based on predetermined routing objectives.

Multiservice Switch node supports the following three rerouting protocols:

- Local rerouting (page 137)
- Global rerouting (page 137)
- Edge-based rerouting (page 138)

Local rerouting is based on the ATM Forum standard domain-based rerouting protocol (DBR). Global rerouting is a Nortel Networks' proprietary extension of the DBR protocol to cover end-to-end rerouting across and entire PNNI network. Local rerouting occurs within predefined regions called local rerouting domains in either a PNNI or HPNNI network. The local rerouting operation is contained entirely within the local rerouting domain. Alternately, global rerouting offers end-to-end rerouting across the entire PNNI or HPNNI network also considered the global rerouting domain.

Attention: When a connection acts as a rerouting or rendezvous node using the PNNI local and global rerouting feature, that connection is not protected during an HSM or FP switchover and as a result is released to the source.

Local rerouting

The local rerouting protocol enables a connection (if registered) to traverse multiple local rerouting domains before reaching the DTL terminator or destination node. Each local rerouting domain controls the rerouting functionality within that domain and each connection segment within a local rerouting domain can be rerouted as an independent connection.

The local rerouting domain boundaries are not tied to the PNNI topology. Multiple local rerouting domains may exist inside a single PNNI peer group. Alternately, a local rerouting domain may include multiple PNNI peer groups. The network operator has the ability to define the local rerouting domain boundaries which are governed by the following guidelines:

- Each node inside a local rerouting domain must be connected by more than one link to other nodes in the local domain. Also, multiple paths must exist to each local domain edge node. This ensures an adequate diversity of routes for connection recovery and path optimization.
- Local domains should be defined to maximize visibility of the physical HPNNI topology for each potential local rerouting node to provide a better selection of routes

The local rerouting protocol defines a local domain link type for a given PNNI link. The link type identifies whether the neighboring node is part of the same local rerouting domain or not. If the link is an intra-domain link, then the PNNI neighbor node is part of the same local rerouting domain. If the link is an inter-domain link, then the PNNI neighbor node is part of another local rerouting domain.

Global rerouting

The global rerouting protocol introduces the concept of a global rerouting domain, which includes all the nodes in a PNNI network. This would also include any and all local rerouting domains. Global rerouting services (connection recovery and path optimization) would operate over all the PNNI inter-domain links between local rerouting domains and/or PNNI intra-domain links.

The global rerouting protocol can also operate in conjunction with the local rerouting protocol on the same connection. This would provide a connection with both local and global connection recovery (protecting both intra-domain and inter-domain PNNI link failures) and path optimization services.

Local global rerouting

The global portion of the local global rerouting protocol behaves in the same manner as the global rerouting protocol. The local portion of the local global rerouting protocol behaves differently than local rerouting in that the local rerouting capability at the local rerouting transit node is determined by the source node which is the local global rerouting node.

Edge-based rerouting

EBR protocol enables connection recovery and path optimization services inside the boundary of the EBR rerouting domain, which encompasses the entire PNNI network. The Nortel Multiservice Switch EBR implementation defines one rerouting domain from the DTL originator to the DTL terminator (end-to-end).

EBR procedures only occur at the edge of a PNNI network. Therefore, unless they are also either the source or termination of a connection, the entry and border exit nodes in a hierarchical PNNI network do not participate in EBR procedures. EBR functionality only takes place within a single PNNI network. When you exit a PNNI network (via IISP or UNI) no EBR signalling is forwarded.

Multiservice Switch supports the ability to combine rerouting protocols and as a result offers five rerouting protocol versions that are listed and defined in the table Rerouting protocol versions (page 138).

Rerouting protocol versions

Rerouting protocol name	Definition
Local and global rerouting	Enables local rerouting capabilities and end-to-end global rerouting capabilities.
Local only	Enables only local rerouting capabilities. This version supports inter-operability with other vendor equipment that supports the DBR protocol.
Global only	Enables only global rerouting capabilities. The connection is subscribed to end-to-end rerouting services only.
Local and EBR	Enables the auto-detection of the rerouting protocol and the differentiation between the EBR protocol and the local rerouting protocol. The rendezvous node determines the rerouting protocol based on the software on the node. If the node has PCR5.1 or later software releases, then the connection will have localOnly subscription capabilities. If the software has pre-PCR5.1 software, then the connection has EBR subscription capabilities. This rerouting protocol enables backward compatibility with the EBR protocol.
EBR only	Enables end-to-end rerouting capabilities.

Rerouting implements two types of rerouting procedures: connection recovery and path optimization. In the event of a network failure, connection recovery recovers failed connections by rerouting the connections without intervention from the end systems (The end system has no knowledge that the network failure or rerouting has occurred.). In the rerouting process, rerouting finds an alternate route for a connection that would otherwise be cleared back to the end-users. Connection recovery is also called hard rerouting or break-before-make rerouting because the incumbent connection segment is released before the establishment of the rerouting connection segment.

Rerouting uses an alarm to notify the network operator which ATM signaling interface has managed connections that were recovered by rerouting procedures. The alarm is set for each ATM signaling interface that manages one or more recovered connections also subscribed to path optimization. Due to the impact on scalability and performance, the information in the alarm is minimal and the alarm is issued only at the interface and not for each connection. This alarm can either be a SET or CLEAR alarm.

For more information on rerouting, see the following sections:

- Connection recovery (page 139)
- Path optimization (page 143)
- Reduced cell loss mechanism (page 149)
- Module optimization passes (page 153)
- Optimization pass combinations (page 154)
- Impact of control processor switchover (page 155)
- Cumulative administrative weight (page 155)
- Rerouting services for different signaling interfaces (page 156)
- Rerouting protocols for different signaling interfaces (page 158)

Connection recovery

Connection recovery (also known as hard rerouting or break-before-make rerouting) provides a failure recovery mechanism for a connections triggered by a network failure in a PNNI routing domain. When a link or a node fails in the network, the network releases the call. If connection recovery has been subscribed to by the connection, the rerouting node blocks the release message and attempts to establish an alternative connection segment to the rendezvous node. The rendezvous node also blocks the release message for the call and waits for the rerouting node to establish an alternative connection segment.

If an alternate route is found, the connection is restored over the new route with the datapath reinstated. Otherwise, the connection is cleared back to its end points through normal call clearing procedures. The end systems have no knowledge that the connection recovery has occurred.

To be able to reroute the connection, the rerouting node must find a route that meets or exceeds the applicable QoS characteristics (that is, a route that has lower numeric values for QoS characteristics) of the ATM service category requested for the original route.

Due to the urgent nature of route failure, connection recovery prevails whenever conflicts with path optimization are encountered. Connection recovery will reroute as many connections as possible at once. In path optimization, connections are rerouted one at a time to conserve network resources.

The following procedure explains how the rerouting process works.

1 Link failure is detected by the network and the incumbent connection segment is released towards the calling and called parties. See the figure Link failure (page 141).

2 The rerouting and rendezvous nodes each receive a RELEASE message for the incumbent connection segment. Both nodes initiate connection recovery procedures by terminating the RELEASE PDUs and starting the rerouting and rendezvous node timers.

3 The rerouting node determines whether there is an alternate satisfactory route to the rendezvous node that meets the QoS requirements for the connection segment. If an alternate path is found, the rerouting node initiates a SETUP message towards the rendezvous node. See the figure Initiating rerouting (page 142). If an alternate path is not found the connection will be released by the rerouting node and the rendezvous node will release the connection after the connection recovery timer has expired.

4 When an alternate route is found (rerouting segment) the rendezvous node receives a SETUP message from rerouting node and switches the datapath from the incumbent connection to the rerouting connection segment.

5 The rendezvous node responds to the rerouting node's call establishment request with a CONNECT message and stops the rendezvous timer.

6 The rerouting node receives the CONNECT message from the rendezvous node.

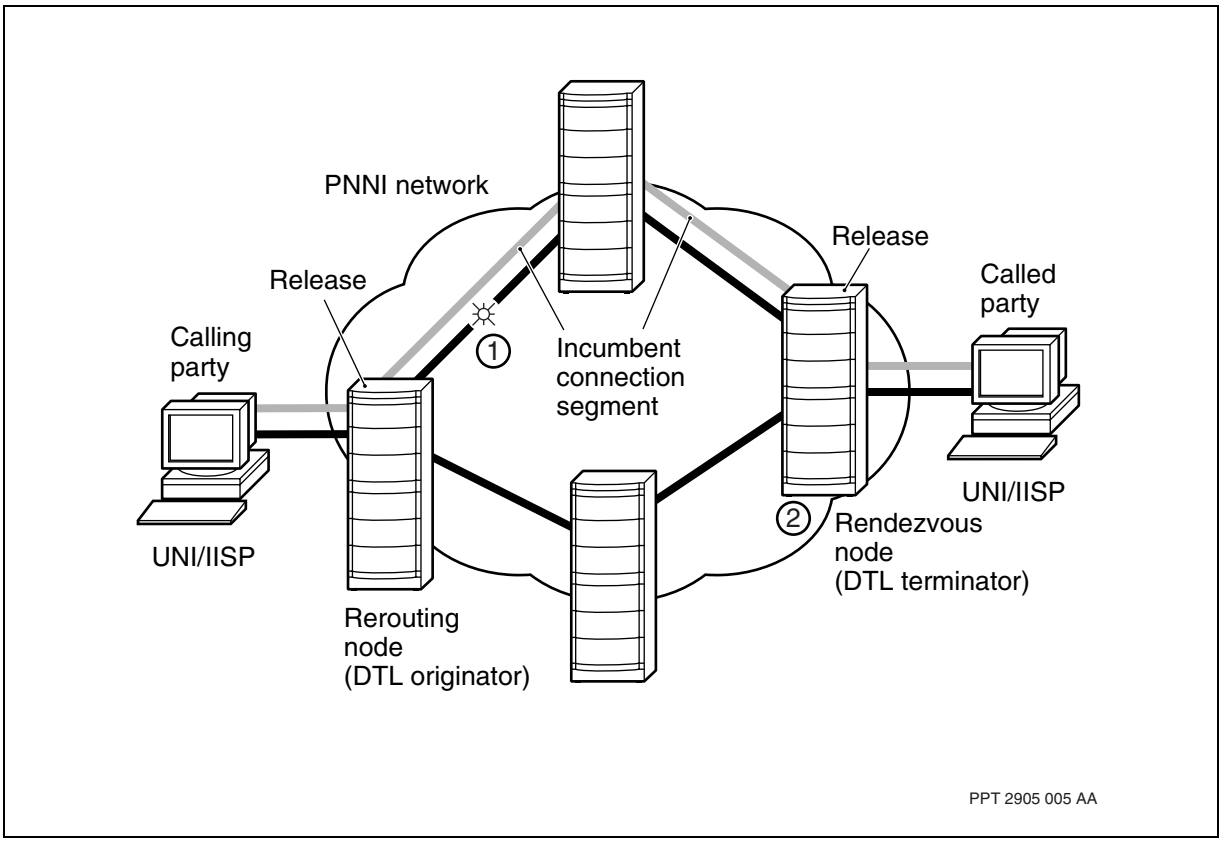
7 The rerouting node switches the datapath from the incumbent connection to the rerouting connection segment, restoring the connection between calling and called parties. The rerouting node then stops the connection recovery timer. See the figure Rerouted connection (page 143).

If network failure occurs to a connection subscribed to both local and global connection recovery services

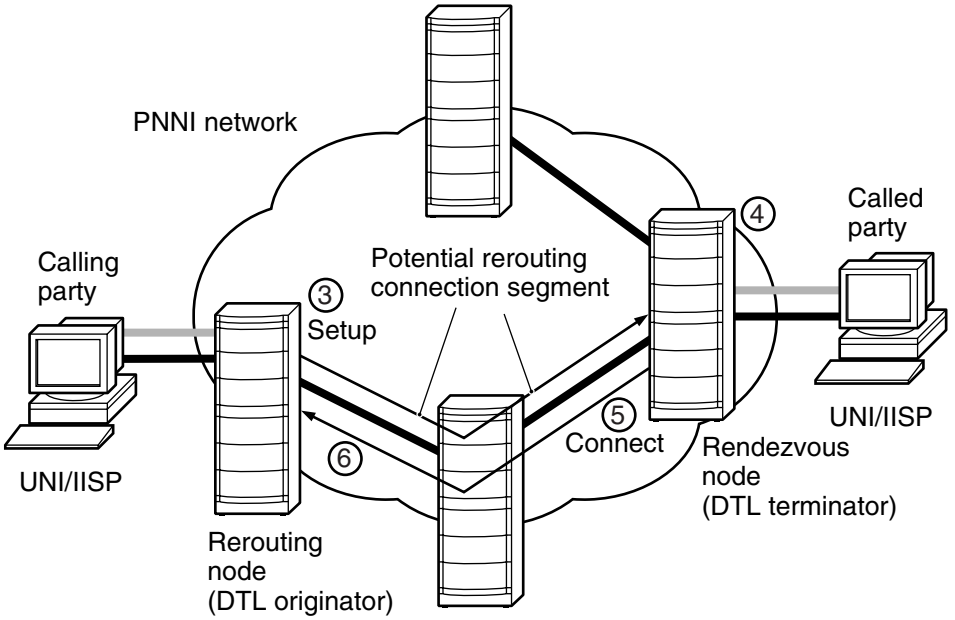
- in the first local rerouting domain (that contains the rerouting node), the connection will attempt a local connection recovery. If the attempt is unsuccessful, a global connection recovery will be invoked.
- outside the first local rerouting domain, the connection will attempt a local connection recovery in that local rerouting domain. If the attempt is unsuccessful, a global connection recovery will be invoked at the global rerouting node.
- outside the global rerouting domain, no global connection recovery is performed. If ever a connection is not able to recover globally, the rerouting node will release the connection in the direction of the calling party number.

Rerouting uses the cumulative QoS parameters of the incumbent connection segment when considering paths for connection recovery (and path optimization). The cumulative values rather than the acceptable values are used because rerouting has no knowledge of the QoS characteristics of the segments of the connection that are outside the PNNI network.

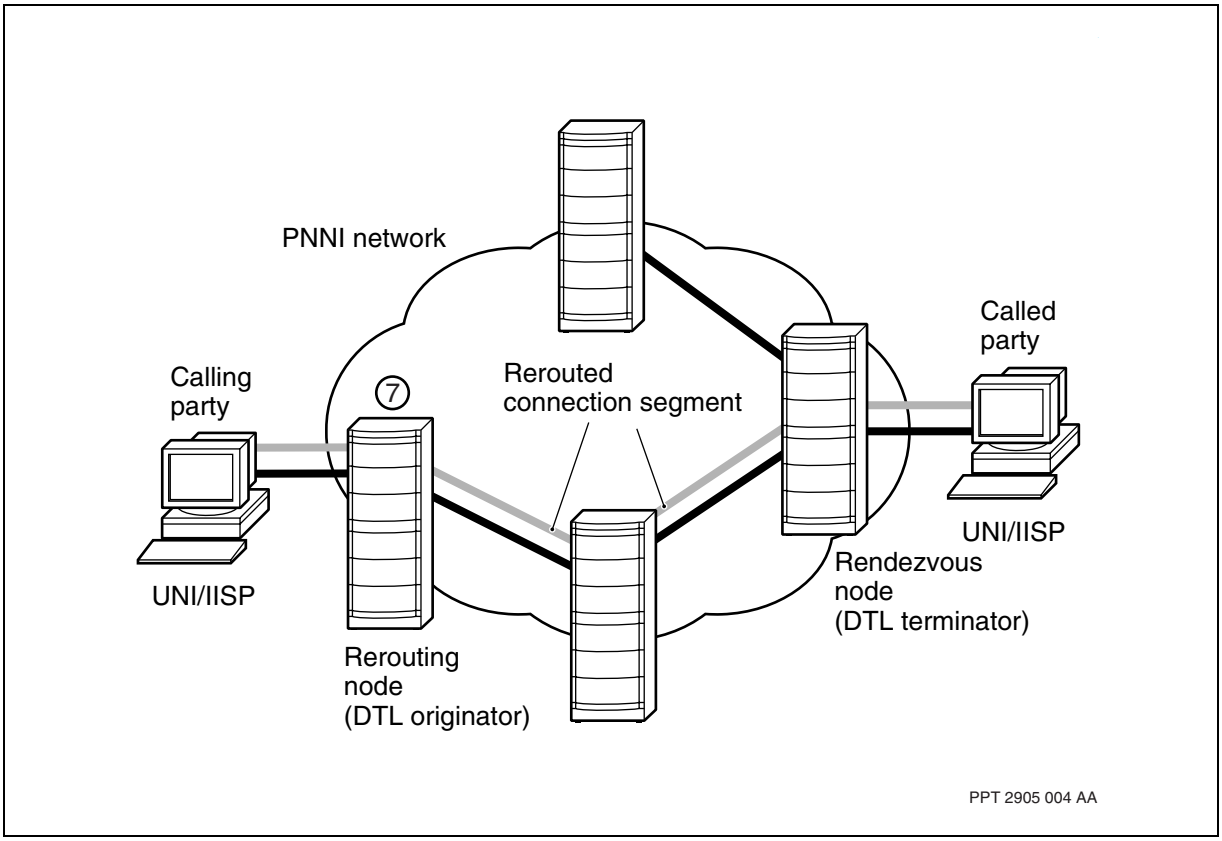
Link failure



Initiating rerouting



Rerouted connection



Path optimization

Path optimization (also known as asymmetrical soft rerouting or make-before-break rerouting) allows active point-to point connections on a PNNI node, an ATM interface, or individual soft PVC and soft PVP connections to be moved to more optimal PNNI routes with minimal disruption in traffic flows. This is achieved by the hot swapping of data paths to a new connection segment after that new connection segment is completely established. This process guarantees cell ordering, does not duplicate cells, and does not drop connections.

When path optimization is triggered, the rerouting node establishes an alternate connection segment to the rendezvous node for subscribed point-to-point connections. When the rerouting connection segment is established (the rerouting node receives a CONNECT message), the rerouting node uses the rerouting connection segment and releases the incumbent connection segment. When the rendezvous node receives the release of the incumbent connection segment, it starts to use the rerouting connection segment.

The only perceivable impact to the connection owner is some cell loss as the incumbent connection is switched to an alternative route during the hot swap interval. Nortel Multiservice Switch nodes offer the option to minimize the amount of cell loss through the reduced cell loss (RCL) mechanism. If the RCL mechanism is not activated, the cell loss is proportional to the time it takes for a RELEASE message to be propagated across the PNNI network. Therefore, if the RCL mechanism is not activated, the period of cell loss can vary and is characterized by the time it takes to clear the incumbent connection segment initiated by the rerouting node. For information on how the RCL mechanism reduces the amount of cell loss, see Reduced cell loss mechanism (page 149).

Regardless of whether or not the RCL mechanism is activated, the clearing of the incumbent connection segment is performed through call control procedures involving the propagation and processing of a Release PDU by all of the PNNI nodes along the incumbent connection towards the rendezvous node.

Path optimizations of active connections are not as critical as the establishment of new calls or the recovery of failed connections. To avoid flooding the network with call establishment requests, Multiservice Switch nodes ensure that there is only one path optimization attempt in progress at an ATM signaling interface at any given time. This practice ensures that network resources are not overutilized during optimization cycles.

The network operator can initiate a path optimization at the connection level, ATM signaling interface level, or the nodal level by using the *optimize* command. When the *optimize* command is issued, the rerouting node determines whether there is a satisfactory alternative connection segment to the rendezvous node. If there is, the rerouting node hot swaps the datapath and clears the incumbent connection segment. For information about the *optimize* command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference* and NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

The set of criteria used for determining whether there is a more optimal connection segment is more selective than that used in the recovery of active connections. When the path optimization command is invoked, the rerouting node calculates a new path optimization metric (AW, maxCTD, or CDV) for each rerouting-capable connection. If the new metric is better than the original path optimization metric, the connection is considered a candidate for path optimization procedures. Note that if the Multiservice Switch node's path load balancing is activated, then the new path must be better than the existing path optimization metric by the variance interval. This helps to ensure that the rerouting connection segment is significantly better. For information on path load balancing and multi-path variance, see Routing using path load balancing methods (page 113).

The following procedure explains how the path optimization process initiates, sets up the connection, and performs the standard method of hot swapping the datapaths.

1 The network operator issues a path optimization command for the ATM signalling interface on the rerouting node. See the figure Initializing optimization (page 146).

2 The rerouting node receives the path optimization command and initiates path optimization procedures for an active connection.

3 The rerouting node identifies an alternate route that exceeds the optimization metric and meets or exceeds the QoS requirements of the incumbent connection. See the figure Setting up the connection (page 147).

When specified path connections with rerouting capabilities are optimized, the node considers the specified path information to determine the optimal path. The primary path is the only path that is considered optimal for these types of connections. Should an optimization result in the connection being re-established on the alternate path or a more preferable automatically computed path, the connection will remain eligible for future optimizations.

4 The rerouting node initiates a SETUP message towards the rendezvous node.

5 The rendezvous node receives the SETUP message and responds with a CONNECT message.

6 The rerouting node receives the CONNECT message from the rendezvous node.

7 Upon receipt of the CONNECT message, the rerouting node hot swaps the data path from the incumbent connection segment to the rerouting connection segment. The rerouting connection segment starts dropping cells in the forward direction at the rendezvous node. See the figure Hot swapping the datapath (page 148).

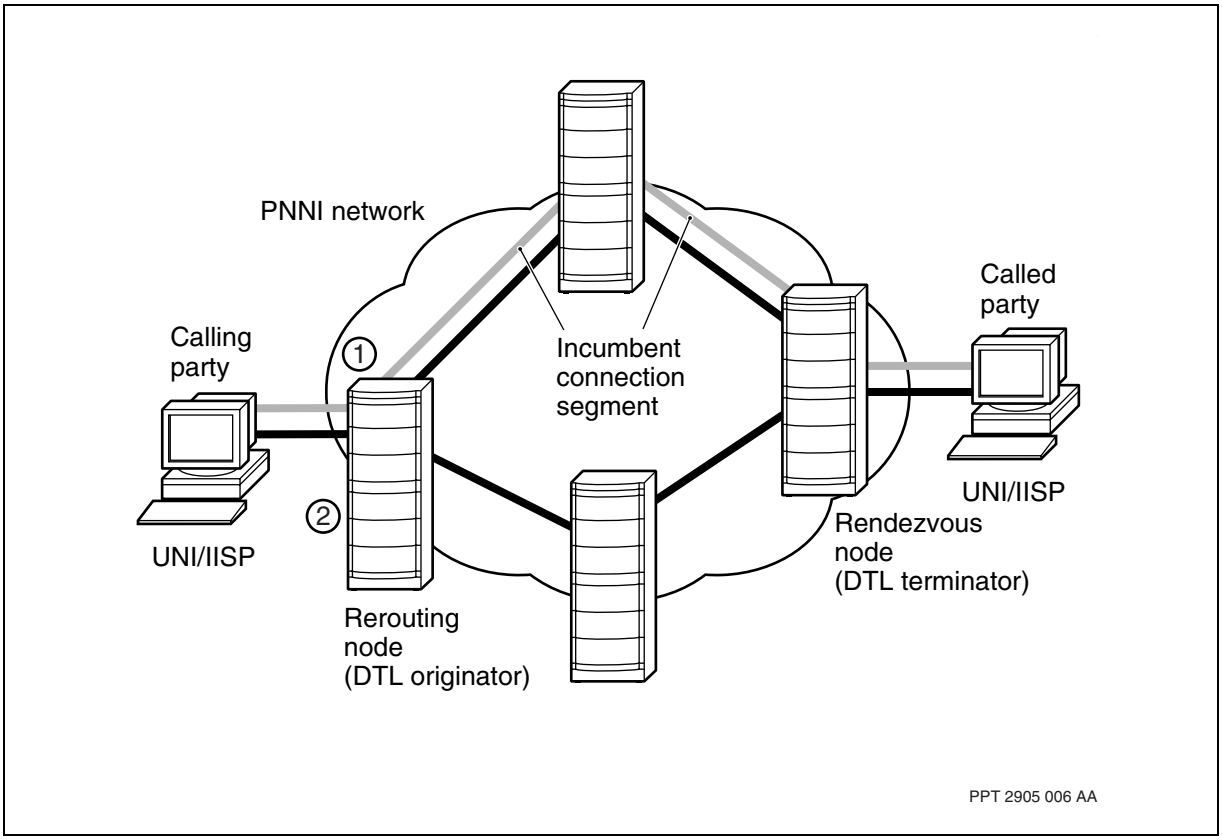
8 The rerouting node releases the incumbent connection, which starts dropping cells in the backward direction at the rerouting node.

9 The rendezvous node receives a RELEASE message on the incumbent connection and hot swaps the incumbent connection to the rerouted connection segment. This process results in the path optimization of the original connection and the restoration of cell relay along the data path in both forward and backward directions. See the figure Optimized connection (page 149).

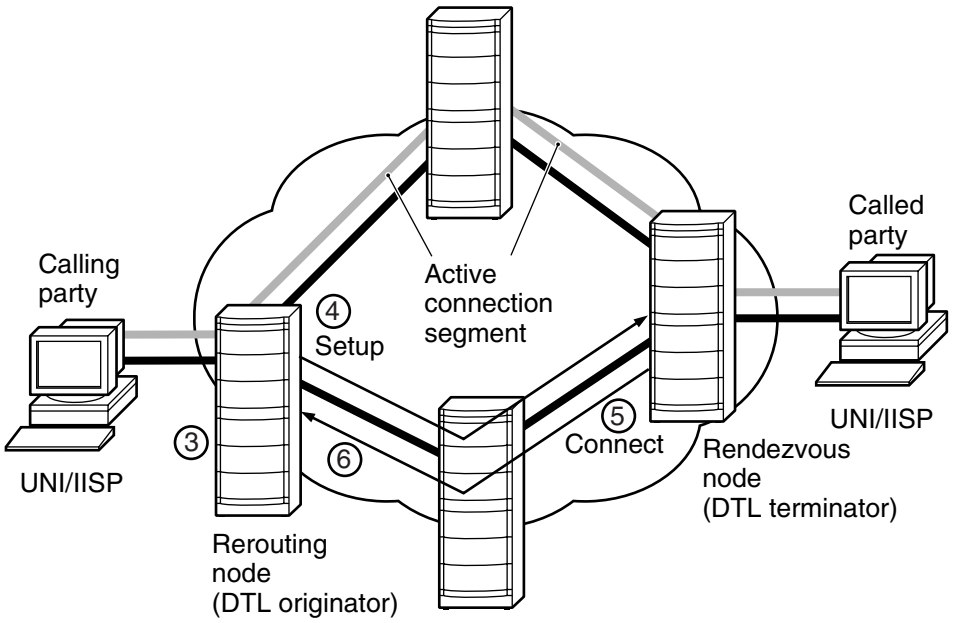
For a connection subscribed to both local and global path optimization services:

- If a path optimization is attempted in the first local rerouting domain (contains the rerouting node), the connection will attempt a global path optimization. If the attempt is unsuccessful, a local path optimization will be attempted.
- If a path optimization is attempted outside the first local rerouting domain, only a local path optimization will be attempted.

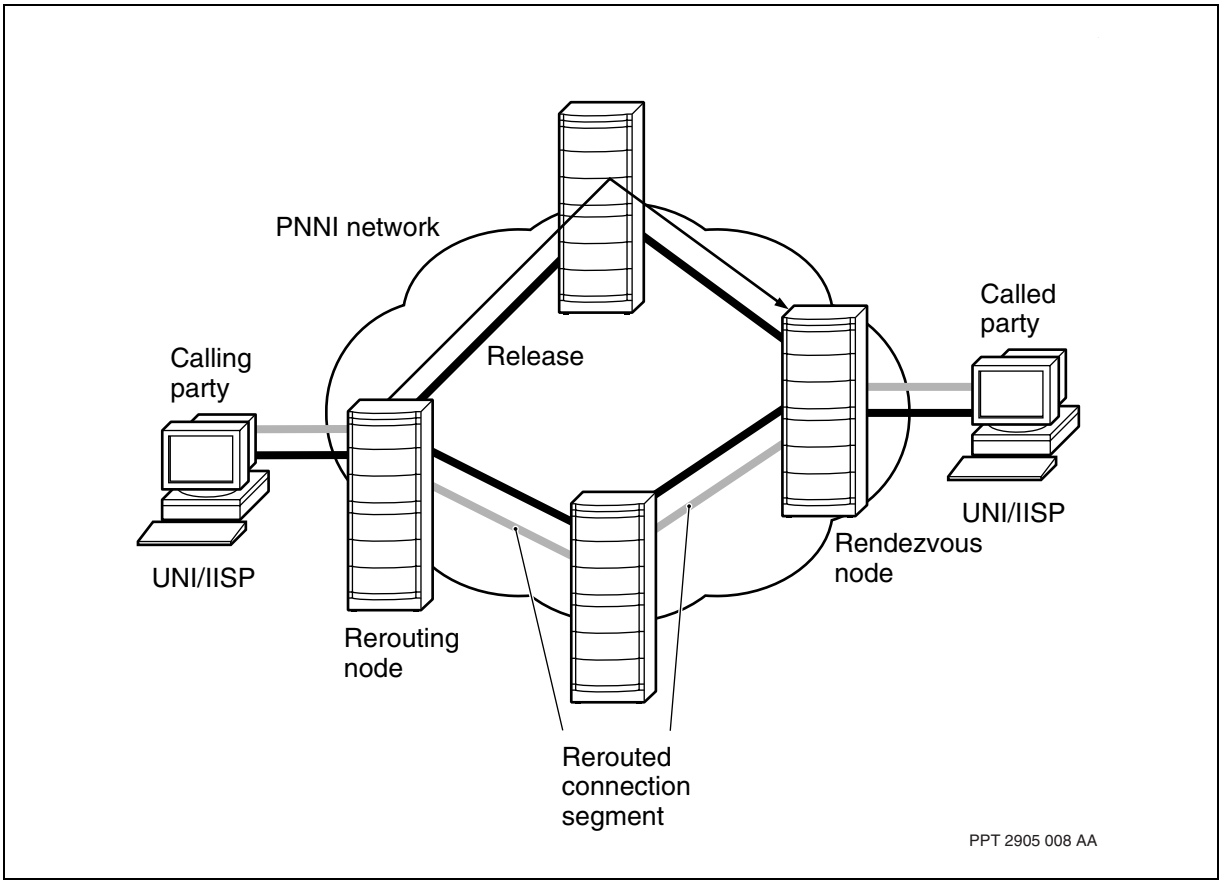
Initializing optimization



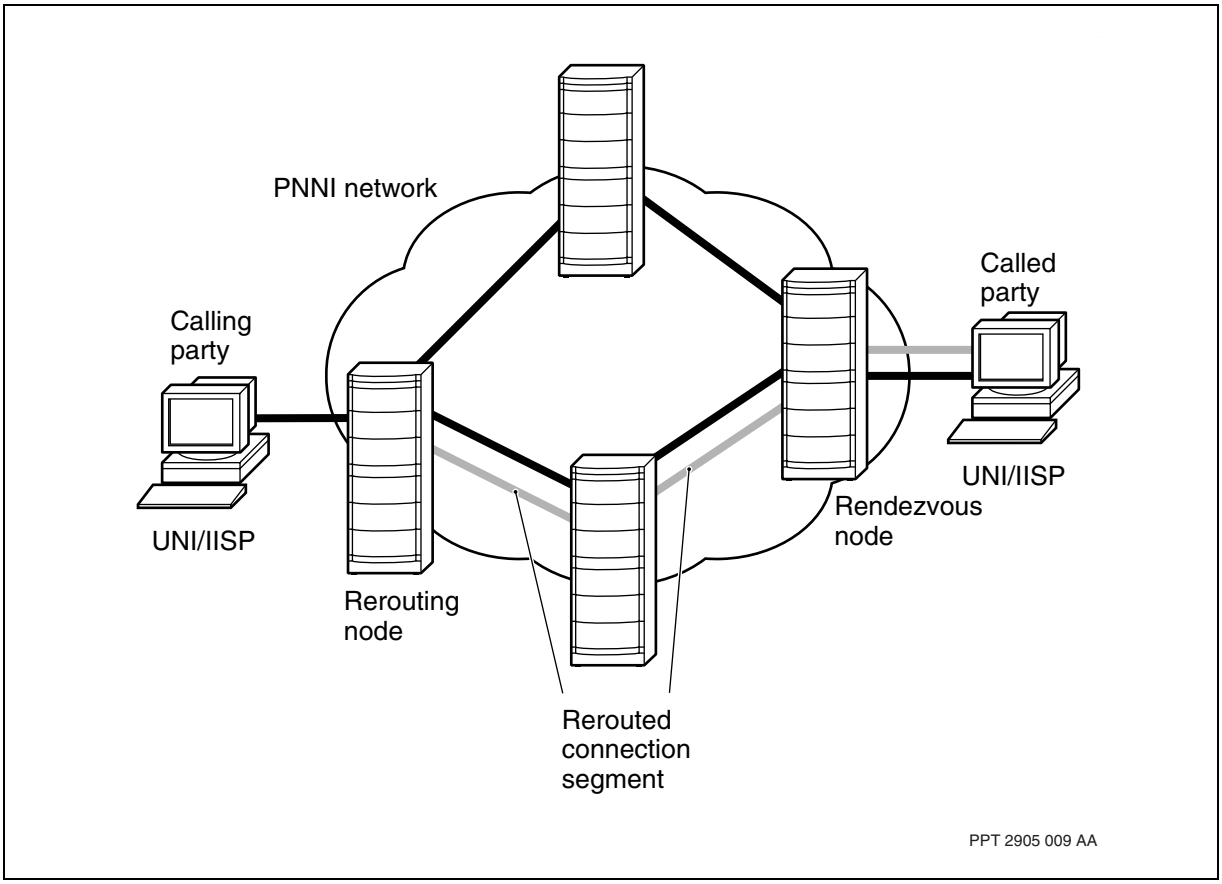
Setting up the connection



Hot swapping the datapath



Optimized connection



Reduced cell loss mechanism

The rerouting reduced cell loss (RCL) mechanism operates to reduce cell loss during path optimization by using a proprietary segment OAM cell and the placement of OAM boundaries to

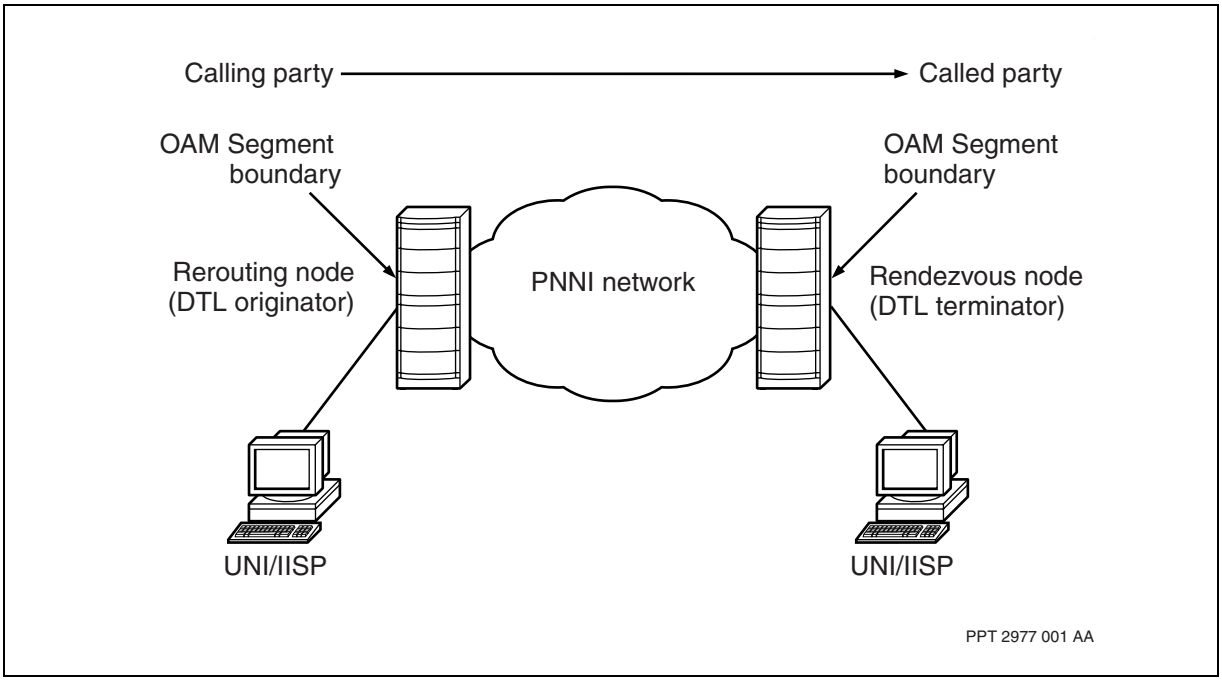
- mark the end of transmission (EOT) on the incumbent connection segment
- trigger the data path swap at the rendezvous node.

The RCL mechanism thus separates the notification of the end of transmission from the data path swap. The cell loss during the data path swap is proportional to the trip delay of the incumbent connection segment and is less than that of the mechanism used by the path optimization procedure based on the ATM Forum document, Domain Based Rerouting. Note that the release of resources across the network is still handled by a RELEASE message.

To implement the RCL mechanism, a new proprietary OAM connection management cell called the End of Transmission (EOT) OAM cell was introduced. The EOT OAM cell is inserted on a per connection basis.

The EOT OAM cells are extracted only at the Connection EndPoints and Segment EndPoints, restricting the rerouting nodes to be set at the Segment OAM Boundary nodes. This constraint means that Segment OAM boundaries cannot exist between the reroute and rendezvous nodes. In other words, there can be no Segment boundaries within the PNNI network. If there is a connection on an OAM segment boundary within the PNNI network between the reroute and rendezvous nodes, then the cell will be extracted and never reach the rendezvous node. See the figure Alignment of the OAM Segment boundary within the PNNI network edge (page 150). The RCL mechanism can only be applied to global and EBR subscribed connections.

Alignment of the OAM Segment boundary within the PNNI network edge



When operating in multi-vendor PNNI environments, perform inter-operability testing to ensure that the OAM cells are handled properly by third party equipment.

For performance monitoring information on rerouting, see NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

The following procedure describes how the RCL mechanism hot swaps the data path at the rerouting node. For more information on the RCL mechanism, see Reduced cell loss mechanism (page 149).

1 Upon rerouting node's receipt of the CONNECT message, the RCL data path swap is triggered. See the figure RCL data path swap at the rerouting node (page 152).

2 The rerouting node then hot swaps the data path for the forward direction from the incumbent connection segment to the rerouted connection segment. Traffic sent in the forward direction is discarded at the rendezvous node until the switchover occurs at the rendezvous node.

3 The rerouting node inserts the End of Transmission (EOT) OAM (operations, administration, and maintenance) cell into the incumbent connection segment's cell stream. The insertion of the EOT cell marks the end of the user data on the incumbent connection segment. Cells will be lost in both directions until the rendezvous node receives the EOT cell and the data path is swapped.

4 Immediately after inserting the EOT OAM cell, the rerouting node sends a RELEASE message to tear down the incumbent connection segment through normal call control procedures.

The following procedure describes how the RCL mechanism hot swaps the data path at the rendezvous node:

1 Upon the receipt of the EOT OAM cell on the incumbent connection, the RCL data path swap at the rendezvous node is triggered. See the figure RCL data path swap at the rendezvous node (page 153).

2 The rendezvous nodes then hot swaps the data path in both directions from the incumbent connection segment to the rerouted segment. Cell flow is reinstated in both the forward and backward directions.

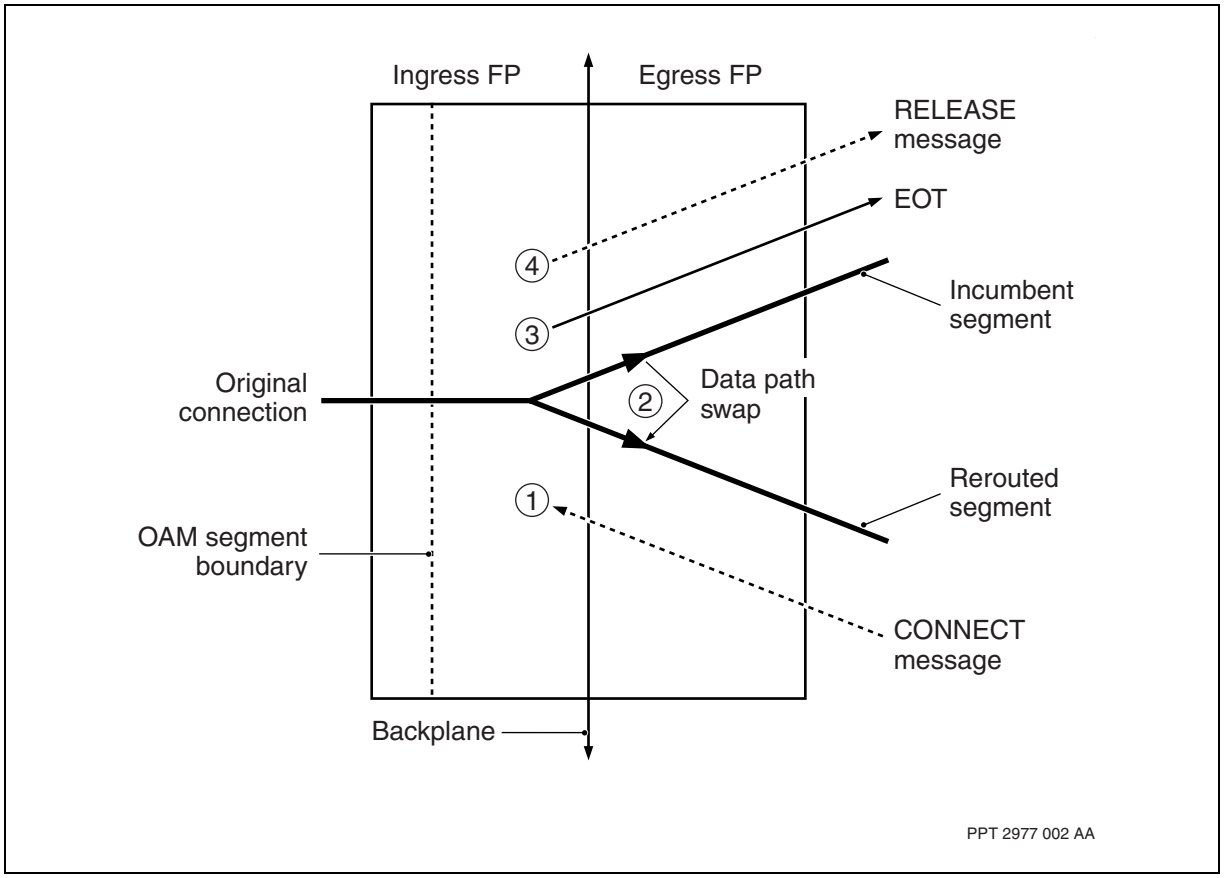
3 The rendezvous node then sends a RELEASE message to tear down the incumbent connection segment.

The ATM layer extracts the EOT OAM cell at the rendezvous node and forwards it to ATM networking to notify call control that the marker indicating the end of user data has arrived. The Nortel Multiservice Switch node performs four checks to ensure that the marker is valid before the rendezvous node switchover procedures are initiated:

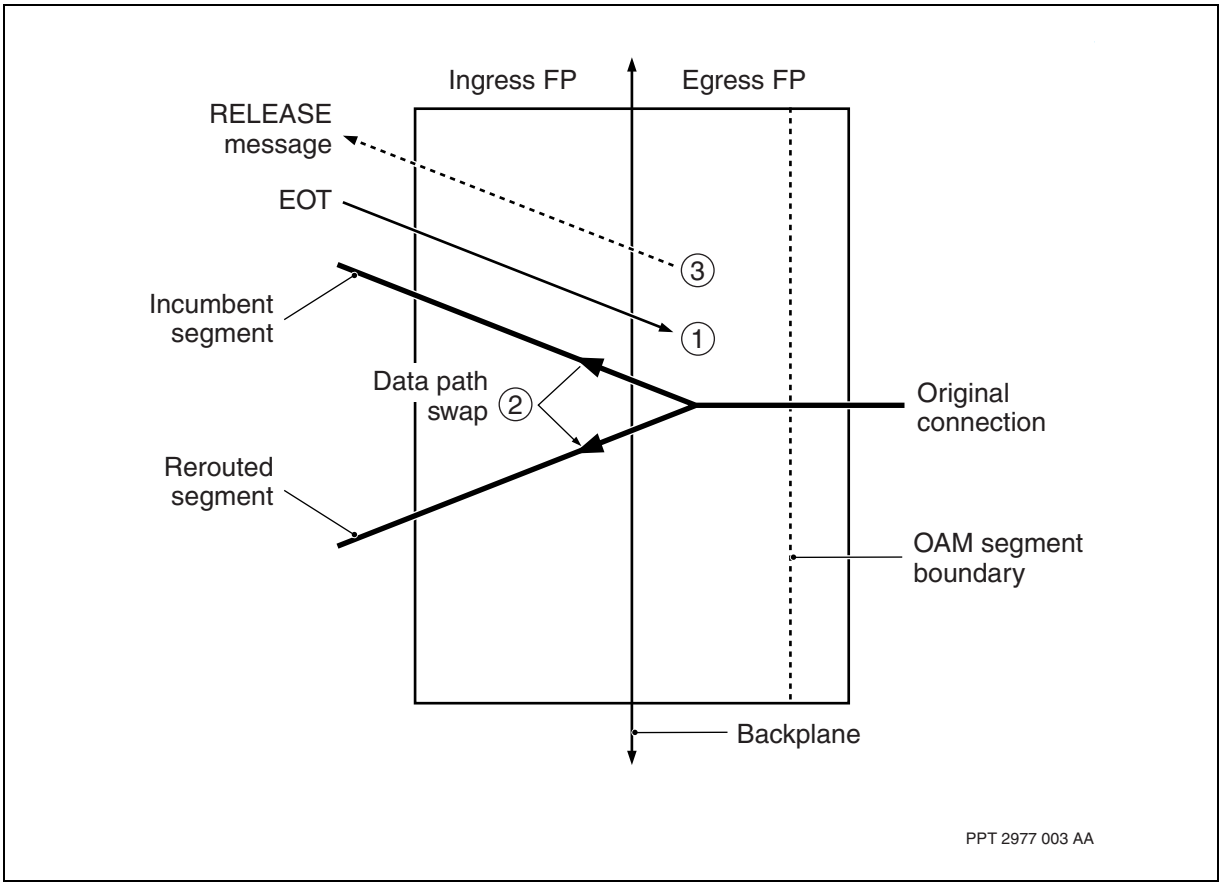
- the connection must be subscribed to rerouting
- connection must be in the Awaiting Switchover state
- the unique Network Call Correlation Identifier (NCCI) encoded in the EOT OAM cell must match the one stored in the connection
- incarnation number encoded in the EOT OAM cell must match the incarnation number currently stored by the connection

If the checks reveal that an invalid EOT OAM cell arrives, no action is triggered and the cell is ignored. If the EOT OAM cell is discarded or extracted by an OAM segment boundary, the arrival of the RELEASE message will trigger the swap of the data path.

RCL data path swap at the rerouting node



RCL data path swap at the rendezvous node



Module optimization passes

Nortel Multiservice Switch nodes provide a time of day path optimization mechanism for the rerouting node to trigger up to a maximum of 12 automated module optimization passes during a 24 hour time period. A module optimization pass attempts to optimize, depending on the setting, either all or only recovered connections subscribed to path optimization services on the rerouting node. Only one module optimization pass can be in progress on a single rerouting node at any one time.

When the time of day optimization mechanism is activated, the ATM routing system triggers a module optimization pass for all rerouting connections that have their rerouting node on the switch. The module optimization pass runs through each interface that has rerouting connections. When the module optimization pass is completed, the ATM routing system waits until the next specified time before triggering another pass.

If the module optimization times are scheduled closely together, the current optimization will be in progress while the next optimization is scheduled to start. In this case, the current module optimization will run to completion and the pending optimization will be skipped. The next scheduled optimization after the one that was skipped will execute at its configured time.

At each rerouting interface, the network operator can specify which connections will be considered during the module optimization pass:

- only recovered connections, or
- all connections subscribed to rerouting, or
- none

Note that a module optimization can be disabled on a per rerouting interface basis. All changes take effect during the next optimization pass.

Optimization pass combinations

The *optimize* command can be used to trigger a manual optimization pass at an individual rerouting connection, interface or against a module. Note that only one module optimization pass can be in progress at any one time.

The manual optimization passes or the time of day optimization passes can be cancelled by using the cancel option with the *optimize* command. If you are cancelling a module optimization that is currently in progress, the module optimization pass will stop as soon as the optimization attempt on the current rerouting interface is completed. If you use the cancel option when there is no optimization pass currently active, the command fails and a notification is sent to the network operator that there was no optimization pass to cancel.

For more information on the *optimize* command, see NN10600-050 *Nortel Multiservice Switch 7400/15000/20000 Command Reference*.

See the table Optimization pass combinations (page 155) for allowed combinations of optimization passes and locations.

Optimization pass combinations

Optimization pass currently in place on the:	New optimization pass attempted on the:	Result
interface	module	• optimization pass on the module is activated • optimization pass on the interface continues uninterrupted: interface being optimized is not considered to be part of the optimization pass on the module
module	module	• new optimization attempt fails and the original module optimization pass in progress continues • network operator is notified that a module optimization pass is currently in progress
interface	interface (other than that in progress)	• optimization pass in progress on the interface continues uninterrupted • new optimization pass on any other interface is activated
interface	interface (same as that in progress)	• new optimization pass on the same interface fails
module	interface	• new optimization pass in progress on the interface is not activated

Impact of control processor switchover

If a module optimization pass is in progress during a control processor switchover, the module optimization pass is interrupted. The interruption is handled as if the module optimization pass was cancelled by the network operator. Any module-wide operational data is cleared.

Cumulative administrative weight

Cumulative administrative weight (CAW) provides an end-to-end administrative weight value for the entire segment for local, global, and EBR connections.

If a routing decision is based on administrative weight (AW), the CAW information is used during optimization to determine if the rerouting segment has a better CAW than the incumbent segment.

CAW is useful in an HPNNI environment. In this case, since the local rerouting node is unaware of the entire HPNNI topology between the local rerouting node and the local rendezvous node, it can not determine, during route calculations, whether the newly calculated route is truly better than the incumbent route. Transport of CAW back to the local rerouting node and storage of the original incumbent CAW value for the incumbent local connection segment enables this comparison to be made. In a flat PNNI network where the local rerouting node is aware of the entire topology, CAW offers no added benefit, since the local rerouting node can determine CAW from topology information.

Each node that is traversed by the connection updates the CAW information. If a node along the path does not support the optional traffic attributes (OTAs) IE then the OTAs IE is discarded. OTAs IE is documented in the following ATM forum specification document: Behavior Class Selector Signalling Version 1.0 af-ca-0159.000. The local rendezvous node and the local rerouting node store the CAW values for each connection subscribed to local rerouting. The global rendezvous node and the global rerouting node store the global CAW values.

Rerouting services for different signaling interfaces

Rerouting services, connection recovery, and path optimization are transported in the rerouting services IE. The term local rerouting services requested data is used to denote fields within the rerouting services IE that indicate a request for either local connection recovery, local path optimization, or both. Similarly, the term global rerouting services requested data will be used to denote fields within the rerouting services IE that indicate a request for either global connection recovery, global path optimization, or both.

If the SETUP message does not contain rerouting services requested data (either it is not populated in the rerouting services IE or the entire IE is not present) this information will be inserted by the first rerouting capable node encountered along the call path. The ingress link into this node must also be an inter-domain link or non-PNNI link. For example, the node is either a local or global domain ingress border node. A node is considered to be rerouting capable if it is provisioned with the feature. The first rerouting capable node will usually be the DTL originator. The types of rerouting services requested are determined by the provisioning of the node or interface.

If the SETUP message contains rerouting services requested data, then the behavior at the ingress link is located in the table Interface and DTL terminator impact on routing services (page 157). The resulting rerouting services requested data that remains after ingress transmission through the link is supported by the node irrespective of the rerouting protocol or rerouting

services provisioned based on the fact that the node receiving the SETUP message is provisioned with the feature and therefore is able to accommodate the request. Any rerouting services which are not supported by this feature (symmetrical rerouting for example) are cleared.

Interface and DTL terminator impact on routing services

Interface type	Behavior	Rerouting service impact/Protocol Impact
Non-DTL terminator:		
IISP, UNI 3.x	Strip the rerouting services IE	-Protocol: not transferred -Local RR services: not transferred -Global RR services: not transferred -EBR services: not transferred
PNNI (intra-domain link)	-Progress the rerouting services IE -Do not clear local or global rerouting services requested data within the rerouting services IE	-Protocol: transparent -Local RR services: transparent -Global RR services: transparent -EBR services: transparent
PNNI (inter-domain link)	-Progress the rerouting services IE -Clear the local rerouting services requested data within the rerouting service IE	-Protocol: transparent -Local RR services: not transferred -Global RR services: transparent -EBR services: transparent
AINI, UNI 4.0	-Progress the rerouting services IE -Clear local rerouting or global services requested data within the rerouting services IE	-Protocol: transparent -Local RR services: not transferred -Global RR services: not transferred -EBR services: not transferred
DTL terminator:		
IISP, UNI 3.x	Strip the rerouting services IE	-Protocol: not transferred -Local RR services: not transferred -Global RR services: not transferred -EBR services: not transferred
UNI 4.0, AINI	-Progress the rerouting services IE -Clear local rerouting services requested data and global rerouting services requested data within the rerouting services IE	-Protocol: transparent -Local RR services: not transferred -Global RR services: not transferred -EBR services: not transferred.

It is possible for rerouting services data to be cleared and inserted multiple times along the call path. Similarly, it is possible for the rerouting services IE to be stripped and inserted multiple times along the call path. This will happen if the call traverses multiple local domains and/or multiple global domains. Multiple global domains are possible when two or more PNNI networks are connected through UNI, IISP or AINI.

Rerouting protocols for different signaling interfaces

The EBR, local, or global rerouting protocols are supported as received in the incoming SETUP message irrespective of the interface (UNI, AINI, or PNNI) receiving the SETUP message. Determination of which rerouting protocol is being used is based on the rerouting services IE and its contents.

If the SETUP message does not contain the rerouting services IE, it will be inserted by the first rerouting capable node encountered along the call path that is either a local or global domain ingress border node. The rerouting protocol requested by this rerouting capable node will be based on the provisioning of the node or interface. In other words, you can have a connection that has different subscription capabilities in each rerouting domain depending on the ingress rerouting signaling interface, for example: AINI or IISP.

Rerouting protocol information is passed transparently across all PNNI, UNI 4.0 and AINI interfaces as seen in the table Interface and DTL terminator impact on routing services (page 157). This means that the entity responsible for inserting the initial rerouting services IE determines the protocol used by the connection segment across all local and global domains. This entity is either the originator of the connection or the first rerouting capable node along the call path.

Rerouting protocol information is not passed across UNI 3.x and IISP interfaces. This means that the first rerouting capable node within the network at the succeeding end of the UNI 3.x or IISP link will determine the rerouting protocol to use based on its provisioning.

UBR with MDCR

Unspecified bit rate (UBR) with minimum desired cell rate (MDCR) enables an end user to assign user plane data to the lower boundary of available bandwidth.

UBR with MDCR is supported on all ATM permanent and switched point-to-point and point-to-multipoint connections on UNI4.0, PNNI1.0 and AINI interfaces. The MDCR information element (IE) is supported to signal the MDCR parameter in SETUP and ADD PARTY messages during call establishment. This MDCR parameter is used to allocate bandwidth to the connections at each interface along the path.

The MDCR IE is not supported on UNI 3.0 and UNI 3.1 interfaces as well as IISP interfaces using 3.0 or 3.1 signaling.

If MDCR is enabled for an existing nailed up UBR connection, the FP card must be reset in order for the configuration change to become active.

Attention: Only nodes with PCR 5.1 or greater software will reserve bandwidth under the *Ca* component. Any nodes with software earlier than PCR 5.1 will still admit the connection, but only as a normal UBR call.

When calls are admitted at the ATM interface, the bandwidth consumed by the minimum cell rate specified in the MDCR is subtracted from the total bandwidth available on the interface. This bandwidth availability is advertised in the PNNI topology. Dynamic route calculations check for bandwidth availability for UBR with MDCR based on the flooded topology data.

UBR with MDCR calls support usage parameter control and traffic shaping and are performed in the same manner as with regular UBR calls.

There are two types of MDCR:

- user specified MDCR
- network generated MDCR

For user specified MDCR, when a Nortel Multiservice Switch node receives a UBR call setup indication with an MDCR that has an origin of user, the call is only admitted if the bandwidth is available. There are checks in both GCAC for routing and ACAC at the interface to ensure that enough bandwidth is available. If the resource checks do not pass, then the call is released with a cause code of 3 (no route to destination) for a GCAC failure or 37 (user cell rate unavailable) for an ACAC failure.

For network generated MDCR, when a Multiservice Switch node receives a UBR call setup indication with an MDCR that has an origin of network, the call is admitted regardless of whether the bandwidth is available. No check is made by GCAC for routing. ACAC at the interface is attempted with the MDCR. But if that fails, then a second attempt is made without the MDCR which then allows the call to establish. This assumes that the interface has not reached its limit of UBR connections.

The MDCR IE is still included in the forwarded UBR call setup indication regardless of whether the MDCR bandwidth requirements are satisfied at an interface. This means that for a UBR call with a network generated MDCR, a Multiservice Switch node performs best effort bandwidth reservation on a per interface basis. The UBR with MDCR call is established as a regular UBR call at the interfaces traversed by the call which can not satisfy the MDCR bandwidth requirements.

Multiservice Switch does not add a network generated MDCR IE. It only forwards a received network generated MDCR IE across AINI and PNNI interfaces and drops it at the egress side of a UNI 4.0 interface.

UBR with MDCR can be configured explicitly using traffic descriptor parameter 9 and implicitly using the default values of the interface. For information on configuring UBR with MDCR, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

Call routing examples

For examples of how a Nortel Multiservice Switch network handles call routing, see the following sections:

- UNI/IISP/AINI static routing examples (page 160)
- PNNI routing examples (page 161)

UNI/IISP/AINI static routing examples

This section provides three examples of static routing under UNI/IISP/AINI.

Example 1: best match with no round-robin load sharing

Port A has primary address 3912334, and port B has primary address 391233.

1 The node receives a call setup request with destination address 391233411111111111112222222223333333333.

2 The node routes the call to port A, since port A has the best match address.

3 If port A is enabled but rejects the call due to insufficient bandwidth, the node does not route the call to port B. Port A presents the best match and the routing algorithm tries only the best match.

4 If port A is disabled, then its static address disappears from the call router table. In this case, the node tries only port B since port B has the best match in the database.

Example 2: best match with round-robin load sharing

Both port A and port B have primary address 3912334 (that is, both ports present an equivalent best match). No other ports present a match.

1 The node receives a call setup request with destination address 391233411111111111112222222223333333333.

2 The first call goes to port A. The next call with the same address match goes to port B. Distribution of call setup attempts continues in round-robin fashion as long as two or more ports are available to take the request.

3 If port A has insufficient bandwidth or rejects the call for any other reason, the node tries port B for that call. If port B also rejects the call, the node clears the call.

Example 3: best match, primary list versus alternate list

Port A has primary address 3912334, port B has alternate address 3912334, and port C has alternate address 3912334.

1 The node receives a call setup request with destination address 39123341111111111111222222222233333333333.

2 The node routes the call to port A, since port A is the best match with a primary address. If port A rejects the call, the node routes the call to port B.

3 The node routes a subsequent call with the same destination address to port A (which rejects the call) and then to port B. If port B rejects the call for any reason, then the node routes the call to port C.

4 If port A also rejects the next call, the node again routes the call to port B. Load sharing does not occur between ports that you configure as alternate.

PNNI routing examples

This section provides two examples of Nortel Multiservice Switch node routing under PNNI.

Example 4: best match, source routing is PNNI

Port A has primary address 3912334. Another node in the same PNNI peer group has Port B with address 39123345.

1 Port A receives a call setup request with destination address 39123345111111111111122222222223333333333.

2 The Multiservice Switch node calculates the source route to Port B.

3 When the node that hosts Port B receives the call setup, it completes a best match and uses the local port.

Example 5: round robin, IISP first, then PNNI

A local port (Port A) on the Multiservice Switch node has primary address 3912334. Two other Multiservice Switch nodes in the same peer group have ports (Port B and Port C) with the primary address 3912334.

1 The node that hosts Port A receives a call setup request with destination address 39123341111111111111222222222233333333333.

2 The node tries to route the call through Port A (the local port) first.

3 On crankback, the node tries the first PNNI node (the node that hosts Port B), but call setup fails.

4 On crankback, the nodes tries the second PNNI node (the node that hosts Port C), but call setup fails.

5 The node that hosts Port A attempts the first and second PNNI nodes in turn until the number of attempts reaches the value that you configured for the *maxReroutesOnCrankback* attribute under the *AtmRouting Pnni* component.

Hierarchical PNNI examples

For general information on how to build a hierarchical PNNI network, see the following sections:

- Creating a new higher level peer group (page 162)
- Creating a new lower level peer group (page 163)
- Creating a new intermediate level peer group (page 163)
- Creating a new physical node at the lowest level (page 163)
- Creating a new physical node at the highest level (page 164)
- Splitting an existing peer group into multiple peer groups (page 164)
- Migrating nodes using a top-down approach (page 165)
- Migrating nodes using a bottom-up approach (page 165)

For detailed information, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management* and NN10600-715 *Nortel Multiservice Switch 7400/15000/20000 ATM Fault and Performance Management*.

Creating a new higher level peer group

There are two common cases requiring the creation of a new higher level peer group:

- when you are preparing to add more physical nodes to the hierarchy
- when a peer group is becoming too large and migration is expected to occur

Creating a new higher level peer group requires you to complete the following tasks:

1 Identify the node at the lower level that is the PGL candidate.

2 For this PGL candidate, add the *PeerGroupLeader* (*Pgl*) subcomponent.

3 Set the Pgl's leadershipPriory (*lPrio*) attribute.

4 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at the higher level.

Creating a new lower level peer group

To add more nodes at a lower level, you will need to create a new lower level peer group. You start by moving a physical node in an existing peer group to a lower level.

1 Identify the physical node that will be moved to a lower level.

2 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at the lower level.

3 For this CfgNode, add the *PeerGroupLeader* (*Pgl*) subcomponent.

4 Set the *Pgl*'s leadershipPriory (*lPrio*) attribute.

Creating a new intermediate level peer group

To add new nodes at a level between a lower level peer group and the higher level containing the LGN representing the lower level peer group, you will need to create a new intermediate level peer group.

1 Identify the lower level peer group.

2 Identify the LGN at the higher level that is representing the lower level peer group.

3 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at a level between the level containing the peer group and the level containing the LGN representing this lower level peer group.

4 For this CfgNode, add the *PeerGroupLeader* (*Pgl*) subcomponent.

5 Set the Pgl's leadershipPriory (*lPrio*) attribute.

Creating a new physical node at the lowest level

Use these tasks to add a new physical node to a peer group where at least one of its neighbors is a physical node.

1 Identify the peer group at the lowest level to which you will add a new physical node.

2 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at the same level as the other nodes in the peer group.

3 Either:

Set the new CfgNode's *peerGroupId* attribute to be that of the peer group.

or:

Set the node prefix as having the same number of significant bits as the other nodes in the peer group.

Creating a new physical node at the highest level

Use these tasks in large PNNI networks to allow physical nodes the ability to take over as the PGL at that level and above in the PNNI hierarchy.

1 Identify the peer group at the highest level to which you will add a new physical node. At least one of existing nodes in the peer group is a LGN.

2 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at the same level as the other nodes in the peer group.

3 Either:

Set the new CfgNode‘s *peerGroupId* attribute to be that of the peer group.

or:

Set the node prefix as having the same number of significant bits as the other nodes in the peer group.

Splitting an existing peer group into multiple peer groups

To divide an existing peer group, perform the following tasks:

1 Identify the peer group that will be divided.

2 For each node you wish to move to the new peer group, you must either:

Set the CfgNode’s *peerGroupId* attribute to be that of the new peer group.

or:

Set the node prefix to have the same number of significant bits as the other nodes in the peer group. If you chose this option, ensure that the node remains in the same higher level peer group. That is, the n-bits of the node prefix (where n is the level number of the higher level) remains as before. Modify the bits between n and m (where m is the level number of the lower level) to uniquely identify the new peer group. The bits in the node prefix after the m bits will uniquely identify the node.

For all nodes that are potential PGLs for the new peer group:

3 Add the*PeerGroupLeader* (*Pgl*) subcomponent at the lower level.

4 Set the *Pgl*’s leadershipPriory (*lPrio*) attribute.

5 Add a new level by configuring a new *ConfiguredNode* (*CfgNode*) component at the higher level.

6 Set the new CfgNode‘s *peerGroupId* attribute to be that of the peer group.

7 To migrate the nodes, see Migrating nodes using a top-down approach (page 165) or Migrating nodes using a bottom-up approach (page 165).

Migrating nodes using a top-down approach

The top-down migration method is the best method to migrate from a single peer group to two peer groups. In this method, you replace existing nodes with LGNs and add new nodes to create new peer groups at the lower levels. This example demonstrates a simple migration.

1 For each peer group you wish to migrate to a lower level, identify the node that you will be replacing with a peer group.

2 For this node, follow the tasks in Creating a new lower level peer group (page 163).

3 To add new lowest level nodes (physical nodes) to the new peer group, follow the tasks in Creating a new physical node at the lowest level (page 163) for each node you wish to add.

Migrating nodes using a bottom-up approach

Use the following tasks for migrating nodes using a bottom-up approach.

1 Identify the node in the peer group that will appear at the highest level in the PNNI network you are creating.

2 For this node, follow the tasks in Creating a new higher level peer group (page 162). These tasks will configure this node as the PGL of the new peer group and will configure a LGN at the highest level.

3 Select another node (not yet migrated) that will be represented as a separate LGN at the same level as the LGN created in step 2.

This node selection must avoid partitioning an existing peer group. Nodes that are part of the same peer group as that selected in step 1 must be selected last.

4 Repeat step 2 for the node identified in step 3.

5 Configure all the hierarchical information for the node selected in step 3.

6 Repeat step 3 to step 5 until you have migrated all the nodes you want to the peer group created in step 3.

Crankback mechanisms

Crankback mechanisms are essential to routing through dynamic networks where transient conditions of congestion or facilities failure may occur. Crankback applies when call setup encounters a node or link that is unable to admit the new connection. Crankback mechanisms let call setup back track from congested or failed areas of the network to establish a new connection through nodes that can support traffic requirements.

Crankback mechanisms are useful for cases in which nodes occasionally become congested. Such congestion can occur due to anomalies in other parts of the network. In areas of the network where nodes are often congested, network re-engineering is required rather than reliance on a crankback mechanism.

For more information on crankback mechanisms, see the following sections:

- Crankback for static routing (page 166)
- Crankback for IISP links between PNNI domains (page 167)
- Crankback for AINI links between PNNI domains (page 170)
- Crankback for dynamic routing in flat PNNI (page 171)
- Crankback for dynamic routing in HPNNI (page 173)
- Crankback for specified paths (page 176)

Crankback for static routing

For connections that use static routing (UNI, IISP, and AINI), a hop-by-hop crankback mechanism provides a means of back-tracking from a failed node in the network. Through this mechanism, the network tries all available primary and alternate routes.

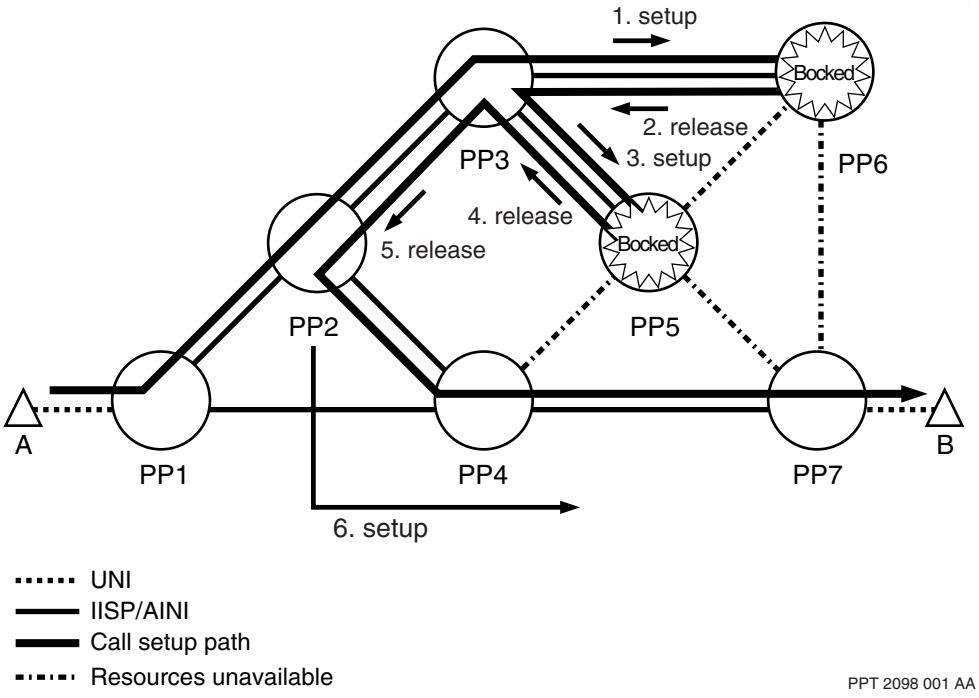
When the node has tried all paths and has failed to find a route, the node sends the call setup request back to the previous node which attempts another route. This process of back-tracking continues until all nodes on a route have tried all possible routes. The figure Crankback example for static routing (page 167) illustrates the crankback mechanism for static routing.

If an interface is found to advertise an address matching the called address, then the call is progressed to that interface. If the interface is not able to accept the call, other interfaces that advertise addresses that match the Called Address are attempted in succession until an interface accepts the call or all the local interfaces are exhausted. The hop-by-hop behavior is configured to specify whether only addresses of equal length to the first one attempted are used or lesser match addresses are also considered. This procedure occurs independently of any crankback indication received across the interface.

In this example, end system A originates a call for end system B. Although the call setup fails at two points, Multiservice Switch node 2 successfully routes the connection through Multiservice Switch nodes 4 and 7 to the destination point.

There is no practical way to detect call routing loops that occur if you improperly configure static addresses. If loops occur, the call eventually clears and unwinds itself after one of the links hits a resource limitation (for example, all VPI.VCI addresses are used or bandwidth is unavailable).

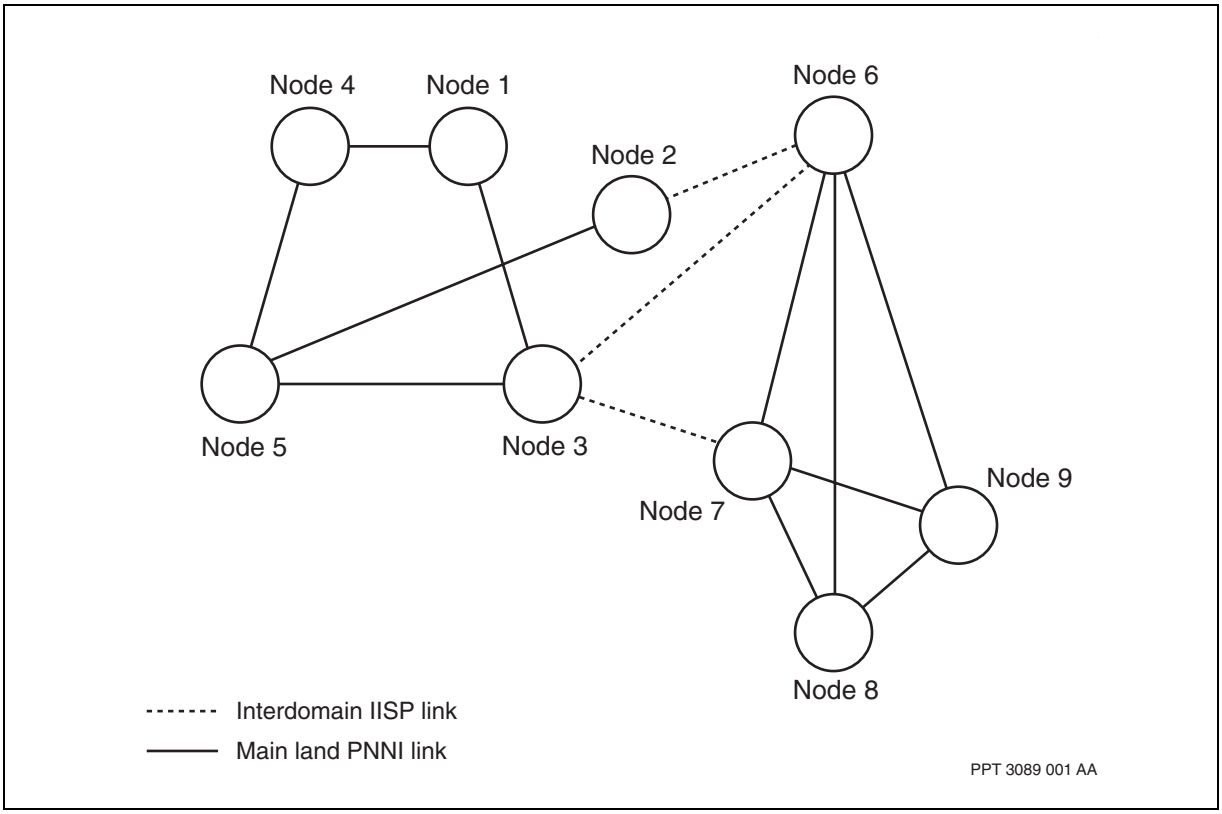
Crankback example for static routing



Crankback for IISP links between PNNI domains

In some network configurations, a few PNNI domains are interconnected by IISP links. Using the figure Network topology (page 168) as an example, the addresses of node 9 are statically provisioned on node 3 using the longest matched address. The addresses of node 9 are also statically provisioned on node 2 using a shorter match address to serve as the alternate routing gateway. As an example of an SPVC from node 1 to node 9, the best route will always be node 1-node 3-node 7-node 9. However, when the failure occurs on the terminating side of the PNNI domain, the originating domain of node 1 is unaware of the failure because no PNNI routing information is sent across the IISP link. Therefore, call attempts will still traverse through node 3 because the routing rule for IISP links is to always route on the longest matched address. The node with the external IISP link connecting PNNI routing domains is called a gateway node from the IISP perspective and subsequently it is also a DTL originator or DTL terminator from the PNNI perspective.

Network topology



The external link crankback feature enhances the existing IISP hop-by-hop crankback by sending the crankback information into the source PNNI routing domain. In this case, the failure to route the call in the terminating PNNI domain will result in a crankback across the IISP link to the gateway. At this point, the gateway node will attempt local rerouting and if there is no available route, this functionality will perform a crankback to the source node. The alternate routing attempt by the source node would choose another IISP gateway rather than using the same IISP gateway. Considering the figure Network topology (page 168), a call from node 1 to node 9 will use the node 3 IISP gateway node because it is the route provisioned with the longest matched address. If the IISP links from node 3 to node 7 and node 3 to node 6 are unavailable, then node 1 would perform alternate routing to the node 2 IISP gateway node. The new call would route through node 6 to node 9 immediately and does not have to activate the SPVC retry timer.

Alternate rerouting is accomplished by the External Link Crankback feature. The behavior of this feature is proprietary and should only be enabled on the call originator, call terminator, DTL originator, DTL terminator, and the IISP gateway nodes where the PNNI domains are interconnected by IISP external links. Nortel Multiservice Switch node call control is modified to include a crankback information element (IE) when hop-by-hop crankback fails at the

DTL terminator. The crankback IE is only inserted when the original failure cause received at the DTL terminator is one of the causes for which a crankback IE is inserted within the PNNI domain. The causes are listed in section 6.4.6.3 of ATM Forum PNNI Specification Version 1.0 af-pnni-0055.000 March 1996 and are summarized in table Cause codes that can generate a crankback IE (page 169). Some cause codes are not supported by Multiservice Switch or are not relevant for failures received across an IISP link. They are noted as such in the table and are not considered during the check for whether to insert a crankback IE. A cause code of 16 for example, which includes SVCs that are released under normal conditions by the user does not appear in table Cause codes that can generate a crankback IE (page 169) because it does not cause a crankback IE to be inserted.

Cause codes that can generate a crankback IE

Cause code	Meaning	Supported on Multiservice Switch
2	The transit network is unreachable.	No
3	The destination is unreachable.	Yes
32	There are too many pending add party requests.	Yes
35	The requested VPCI/VCI is not available.	Yes
37	The user cell rate is not available.	Yes
38	The network is out of order.	Yes
41	A temporary failure has occurred.	Yes
45	No VPCI/VCI is available.	Yes
47	The resource is unavailable and unspecified.	Yes
49	The quality of service is unavailable.	Yes
57	The bearer capability is not authorized.	Yes
58	The bearer capability is presently not available.	No
63	The service or option is not available and unspecified.	Yes
65	The bearer service is not implemented.	Yes
73	The combination of traffic parameters are unsupported.	Yes
128	The next node is unreachable.	PNNI only
160	The DTL transit is not the node ID.	PNNI only

During route calculation, the DTL originator uses the information contained in the crankback IE to exclude the previous DTL terminator for the purposes of terminating the call. The route calculation can still use the node as a transit to arrive at a new IISP gateway. The alternate routing attempt will first look for

static addresses of length equal to the one attempted initially. When no equal length addresses are available, then the next best matched address is used. This is particularly important as the network can be configured with multiple preferred links or to have preferred and backup links based on the address length. With this feature disabled, a Multiservice Switch node considers only the longest match available and would not crankback to a shorter matched address.

Crankback for external link needs to be deployed and activated on a node by node basis. This feature introduces the *useBestMatchAddrOnCrankback* attribute under the *Artg* component and the *interDomainCrankback* attribute under the *AtmIf Iisp* component. The *useBestMatchAddrOnCrankback* attribute designates whether the node would use best match addresses when performing a crankback. The *interDomainCrankback* attribute specifies whether the DTL terminator would insert a crankback IE in the release PDU back to the source node when hop-by-hop routing crankback fails at the IISP gateway node. For more information on provisioning crankback for external link, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

Crankback for AINI links between PNNI domains

A call received over an AINI signalling link that cannot be progressed beyond the receiving node may be subject to crankback and alternate procedures. In particular, the AINI transports crankback indications between PNNI networks.

The following list displays the three basic scenarios where a crankback IE is inserted at an AINI due to insufficient resources:

- on the CP during route calculation (whole node)
- at the egress AINI (preceding end of the following link)
- at the ingress AINI (succeeding end of the previous link)

The blocked transit types used are the following AINI specific types:

- call or party has been blocked at or beyond the succeeding node
- call or party blocked at the succeeding end of the interface

When a release indication is received across an AINI, alternate routing procedures apply. Other static routing interfaces may be tried including UNI and IISP interfaces when ARtg useBestMatchAddrOnCrankback is set on the node. Once all interfaces are exhausted, the call is released. A crankback indicating that the call or party has been blocked at or beyond the succeeding node is inserted if a crankback IE was received.

If the call is cranking back into a PNNI domain, the crankback is translated appropriately. A PNNI Crankback is inserted at the DTL terminator in the direction of the called user if the following conditions are met:

- the first local interface used at the DTL terminator is an AINI
- a release containing a crankback IE is received across that interface or is generated by that interface
- the call can not be progressed any further when static crankback is exhausted

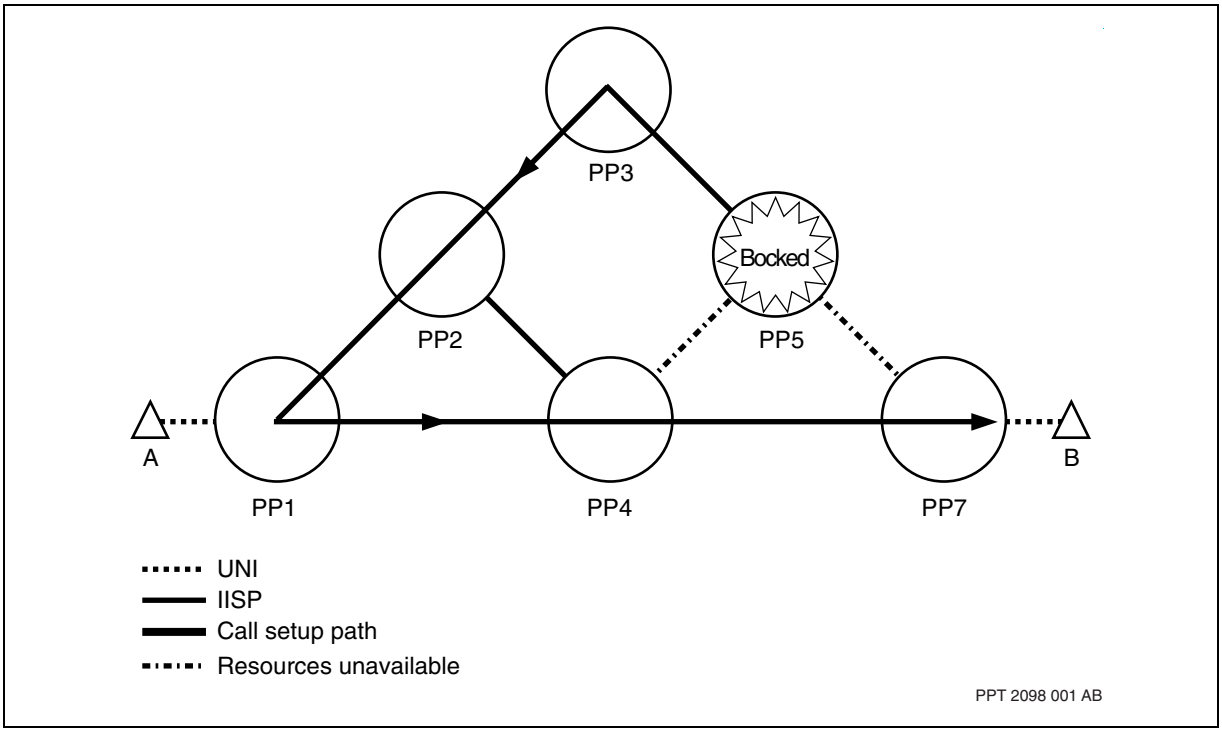
Alternate routing attempts may then be made at the DTL originator or entry border node. The crankback IE inserted in the AINI and sent in the release message into the PNNI network and interpreted as a blocked node terminating crankback. There is a crankback in the release message if there was a crankback in the next PNNI domain on the AINI interface. If the call fails at a UNI, there is no crankback inserted. This means that if the call fails at the UNI in a foreign PNNI domain, then there is no attempt at alternate routing in the domain that originated the call.

Crankback for dynamic routing in flat PNNI

For connections that use dynamic routing, the crankback mechanism provides a means of back-tracking from a failure in the network during call setup. This mechanism lets the source node attempt to alternately route the call based on the address advertisement via another link. In other words, the address is reachable through a different link. In a flat PNNI network, the source node or DTL originator is the node that is responsible for crankback procedures.

The figure Crankback example for flat PNNI (page 172) illustrates the crankback for dynamic routing in a flat network.

Crankback example for flat PNNI



When calls are routed across a PNNI network, the route to be taken is transported as a designated transit list (DTL) stack which is composed of one or more DTL information elements (IEs). A crankback information element (IE) is used during call establishment procedures to indicate to the designated transit link (DTL) originator and border nodes that the progress of the call setup was blocked and crankback procedures can be initiated. The IE indicates which node or link cannot accept the call or connection, the level of the PNNI hierarchy at which the crankback is being carried out, and the reason why the failure occurred. The DTL originator and border nodes use this information to calculate an alternate route across the network or peer group to avoid nodes and links that are unusable.

The following list contains the three different categories of crankback failure:

- The first category covers failures within the PNNI domain which may be subject to crankback.
- The second category is for calls that progress to and are rejected by the called user. Calls that reach the called user are released all the way back to the calling user.
- The third category is for calls that fail when the DTL terminator determines that the UNI to the called user cannot carry the call. In this case, the call may be cranked back and alternate routing within the PNNI cloud is attempted. A crankback IE is inserted when the called party address can

not be found at the DTL terminator. Otherwise, we abandon the crankback.

The table Cause codes that can generate a crankback IE for flat PNNI (page 173) contains a list of all the cause codes that can generate a crankback IE and their meanings.

Cause codes that can generate a crankback IE for flat PNNI

Cause code	Meaning	Supported on Multiservice Switch
2	The transit network is unreachable.	No
3	The destination is unreachable.	Yes
32	There are too many pending add party requests	Yes
35	The requested VPCI/VCI is not available.	Yes
37	The user cell rate is not available.	Yes
38	The network is out of order	Yes
41	A temporary failure has occurred.	Yes
45	No VPCI/VCI is available.	Yes
47	The resource is unavailable and unspecified.	Yes
49	The quality of service is unavailable.	Yes
57	The bearer capability is not authorized.	Yes
58	The bearer capability is presently not available.	No
63	The service or option is not available and unspecified.	Yes
65	The bearer service is not implemented.	Yes
73	The combination of traffic parameters are unsupported.	Yes
128	The next node is unreachable	Yes, on PNNI only
160	The DTL Transit is not my node ID	Yes, on PNNI only

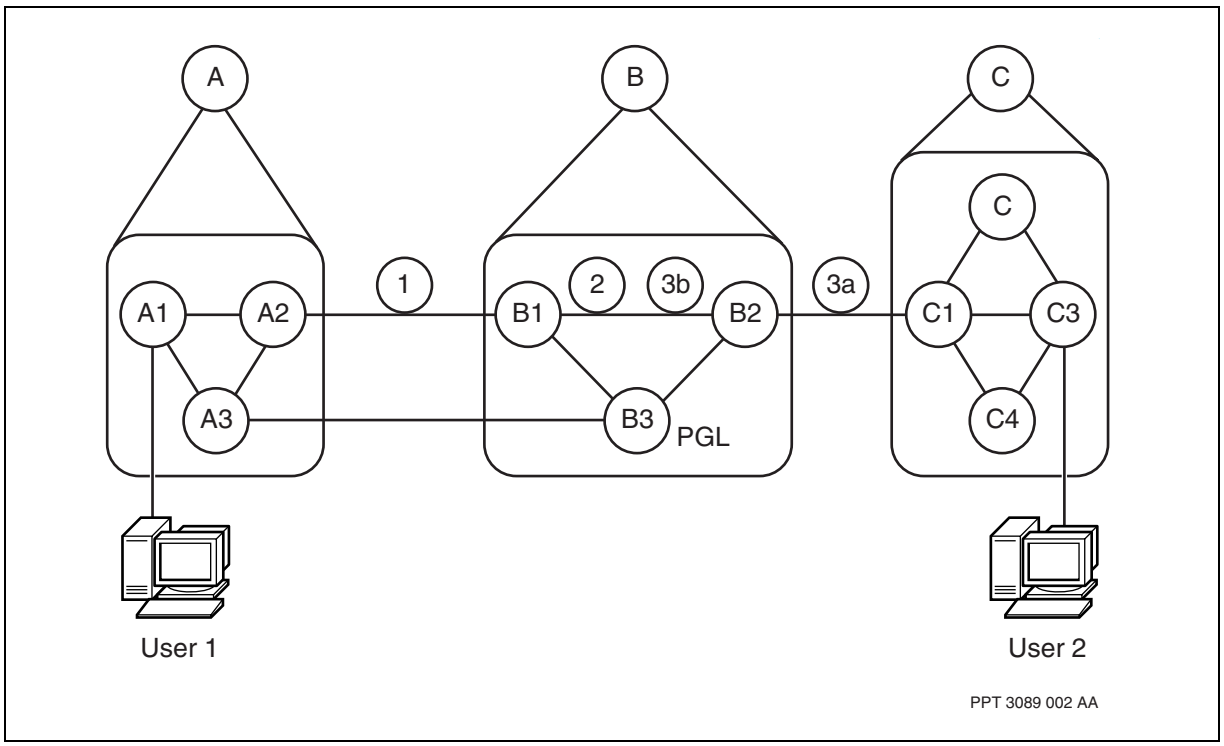
Crankback for dynamic routing in HPNNI

For connections that use dynamic routing, the crankback mechanism provides a means of back-tracking from a failure in the network during call setup. In a hierarchical environment, crankback procedures are handled differently. When a call setup enters a new peer group the entry border node retains a copy of the setup message specifically for crankback purposes. For example, in Crankback example for dynamic routing for HPNNI (page 174) if a failure occurs between peer groups (1), then the call is cranked back to the preceding peer group's entry border node (A1). If a failure occurs within an intermediate peer group (2), then the call is cranked back to the entry border node of that

peer group (B1). If a failure occurs either within (3b) or between peer groups (3a) and the entry border node (B1) does not have the ability to alternately route the call, then the call will be cranked back to the preceding peer group. Therefore, the potential exists for the call to be cranked back all the way to the DTL originator or the source node (A1) of the call setup.

The first eligible border node that is closest to the point of blocking, must regenerate an alternate path consistent with the original route. Alternate routing is required in the cases where calls are rejected along the path due to local CAC performed by a node.

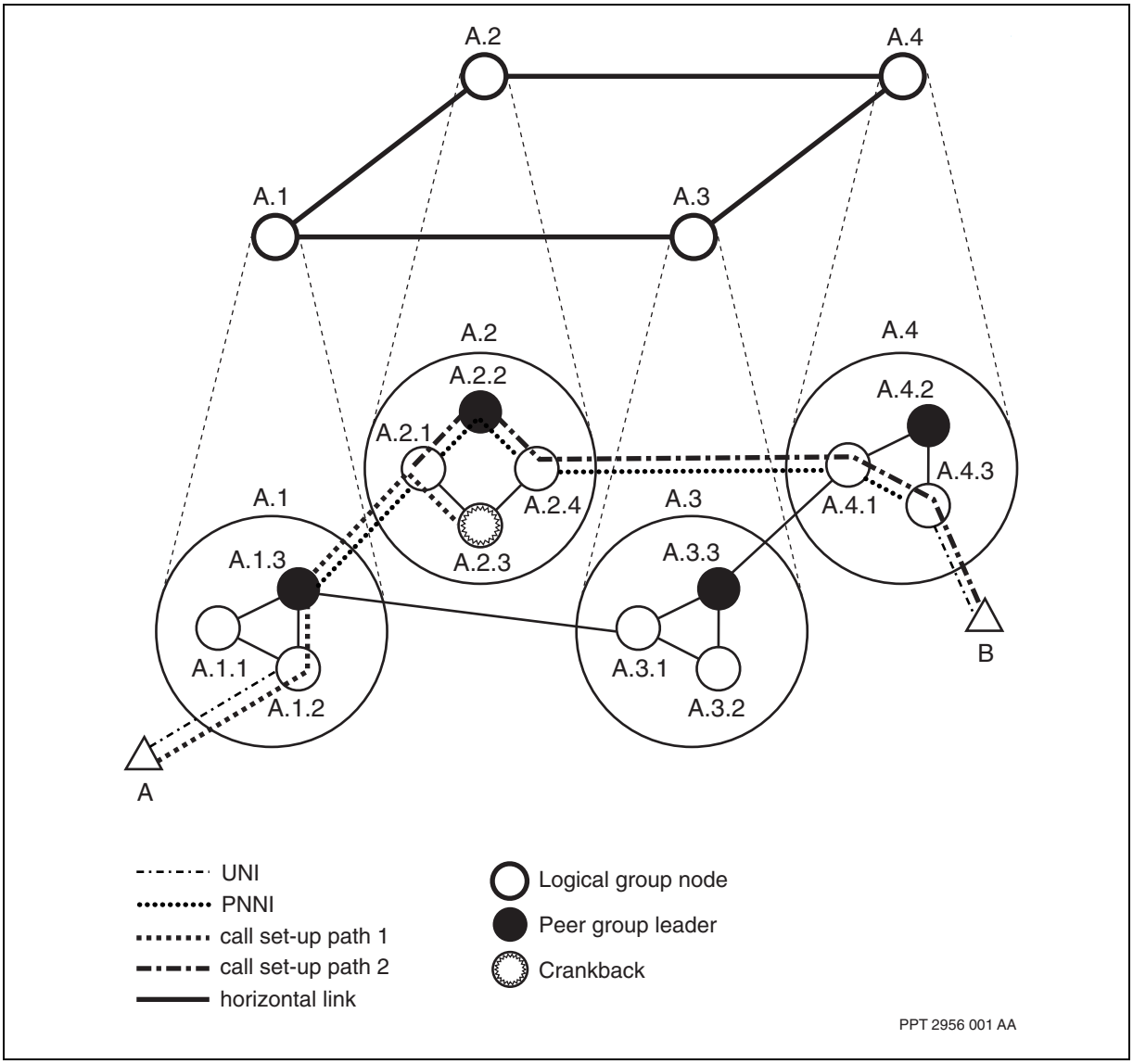
Crankback example for dynamic routing for HPNNI



The entry border node has certain responsibilities that make it act like a DTL originator (for example, expanding the DTL across its own peer group). Thus, the entry border node of a terminating peer group always calculates a path to the destination address.

The figure Crankback example for dynamic routing for HPNNI (page 174) illustrates the crankback for dynamic routing in a hierarchical network.

Crankback example for HPNNI



The table Cause codes that can generate a crankback IE for HPNNI (page 175) contains a list of all the cause codes that can generate a crankback IE and their meanings.

Cause codes that can generate a crankback IE for HPNNI

Cause code	Meaning	Supported on Multiservice Switch
2	The transit network is unreachable.	No
3	The destination is unreachable.	Yes

(1 of 2)

Cause codes that can generate a crankback IE for HPNNI (continued)

Cause code	Meaning	Supported on Multiservice Switch
32	There are too many pending add party requests	Yes
35	The requested VPCI/VCI is not available.	Yes
37	The user cell rate is not available.	Yes
38	The network is out of order	Yes
41	A temporary failure has occurred.	Yes
45	No VPCI/VCI is available.	Yes
47	The resource is unavailable and unspecified.	Yes
49	The quality of service is unavailable.	Yes
57	The bearer capability is not authorized.	Yes
58	The bearer capability is presently unavailable.	No
63	The service or option is not available and unspecified.	Yes
65	The bearer service is not implemented.	Yes
73	The combination of traffic parameters are unsupported.	Yes
128	The next node is unreachable	Yes, on PNNI only
160	The DTL Transit is not the node ID	Yes, on PNNI only

(2 of 2)

Crankback for specified paths

If a specified path originator receives crankback information during call setup on a primary or alternate path, the crankback information will be ignored. That is, the information will not be transmitted to the route cache to remove paths that contain nodes or links where crankback occurred. Also, the blocked entity is excluded from the next specified path. As a result, the crankback information can not be used to update the current topology because the DTL of the SETUP was not generated by the topology.

However, if a specified path originator receives crankback information during call setup using a path chosen by PNNI routing selection, then the crankback information will be used. That is, the information is transmitted to the route cache to remove paths that contain nodes and links at the crankback and the blocked entity will be excluded from the next specified path.

For the purpose of counting the number of crankback attempts (a number used to limit crankback retries), the manual paths are ignored in this count. That is, the first automatic route selection is considered the first attempt for crankback retry purposes.

If an established call is released due to network failures, an immediate call setup will occur. If this call setup fails, then the retry mechanism for SPVC/P is initiated. In this case, the specified path connection will behave as if it were a new call by trying the primary path first, followed by the alternate path, if provisioned, and finally the automatic PNNI route selection, if provisioned.

Call routing for point-to-multipoint connections

Nortel Multiservice Switch nodes forward call setup requests across the network using both static routing and dynamic route discovery. The routing protocol depends on the configured interface: static routing for UNI, IISP, and AINI interfaces, and dynamic route discovery for PNNIs.

For more information on point-to-multipoint connections, see Point-to-multipoint connections (page 179).

For more information on call routing for point-to-multipoint SVCs and SPVCs, see the following sections:

- Static routing (page 177)
- Dynamic routing (page 177)

Static routing

Point-to-multipoint routing of Setup and Add Party messages at the UNI, IISP and AINI interfaces uses the existing static routing capability between the root and the leaves of the multicast tree. For further details on UNI and IISP routing, see Static routing (page 95).

Dynamic routing

Point-to-multipoint routing of Setup and Add Party requests across the PNNI interface uses existing dynamic route discovery between the root and the leaves of the multicast tree. The PNNI route computation permits the optimization of PNNI routing parameters from a point-to-point perspective only.

Optimal point-to-multipoint call routing in which the branch occurs at the last common node (LC node) closest to the leaf requires an optimized multicast tree. Nortel Multiservice Switch nodes do not support an optimized multicast tree. As a result, the call routing algorithm does not ensure the minimum amount of branching for point-to-multipoint routing and GCAC does not account for the existing multicast tree when calculating the optimal path.

Routing considerations for virtual interfaces

The routing system does not distinguish between actual interfaces (UNIs, IISPs, and PNNIs) and virtual interfaces. Nortel Multiservice Switch nodes apply the same routing algorithm for both.

Routing behavior of spared ATM interfaces on Multiservice Switch

For Nortel Multiservice Switch 15000 and Multiservice Switch 20000, the ATM routing application can be spared as a hot standby application.

Hot standby applications and features offer hitless services during an equipment switchover. Hot standby applications and features incur minimal traffic interruptions and established connections stay up.

Equipment switchovers can occur because:

- an active card fails
- a hitless software migration is in progress

The behavior of ATM routing is described in Behavior of Multiservice Switch node switched ATM services during an FP switchover (page 30).

See NN10600-550 *Nortel Multiservice Switch 7400/15000/20000 Common Configuration Procedures* for a full description of hitless services and hot, warm and cold standby applications and features.

Point-to-multipoint connections

Nortel Multiservice Switch point-to-multipoint connections efficiently distribute identical information from one subscriber to many selected subscribers. The point-to-multipoint functionality provides unidirectional multicast VCC-based connections.

Navigation

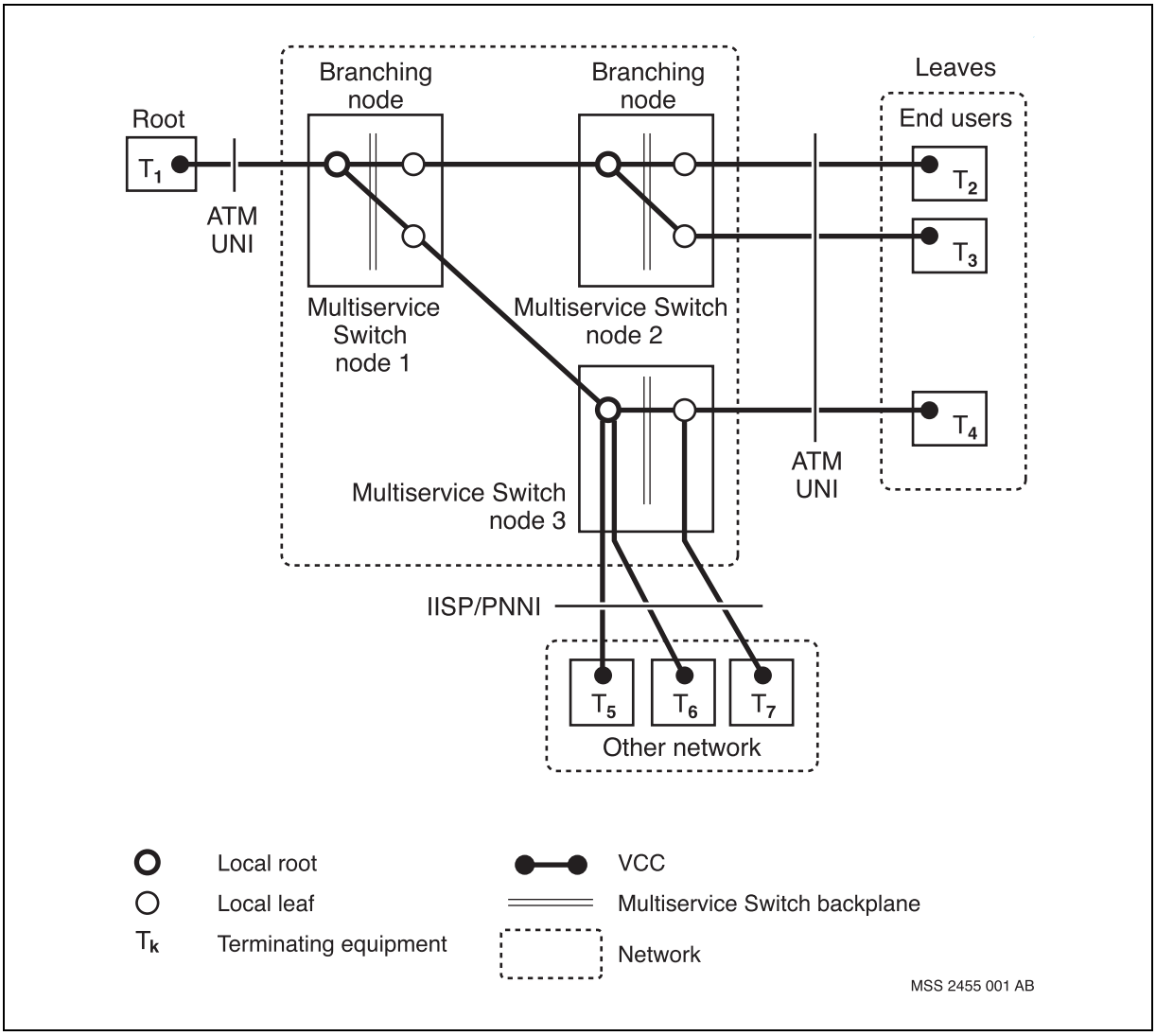
Overview of point-to-multipoint connections

In a point-to-multipoint implementation, the network multicasts information unidirectionally from one caller (the root) to a specific set of called parties (the leaves). The root and leaves are terminating equipment at the edge of the ATM network. At network nodes within the ATM cloud, the node links calls from a receive connection (the local root) to a transmit connection (the local leaf). Network nodes at which a local root links to two or more local leaves are called branching nodes.

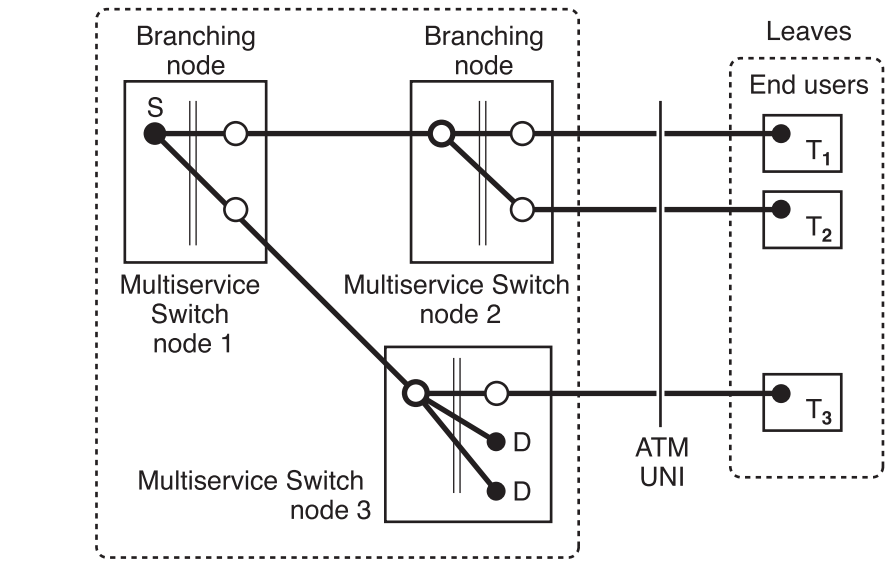
The following figures display point-to-multipoint connections:

- Multiservice Switch point-to-multipoint SVCs (page 180) displays a high-level view of Nortel Multiservice Switch point-to-multipoint SVCs.
- Multiservice Switch point-to-multipoint SPVCs (page 181) displays a high-level view of point-to-multipoint SPVCs.

Multiservice Switch point-to-multipoint SVCs



Multiservice Switch point-to-multipoint SPVCs



Local root
Local leaf
T_k Terminating equipment
S Source point of PMP SPVC
D Destination point of PMP SPVC

VCC
Multiservice Switch backplane
Network

Note: Multiservice Switch node 3 displays examples of logical multicasting used to support multiple branches to the same egress port

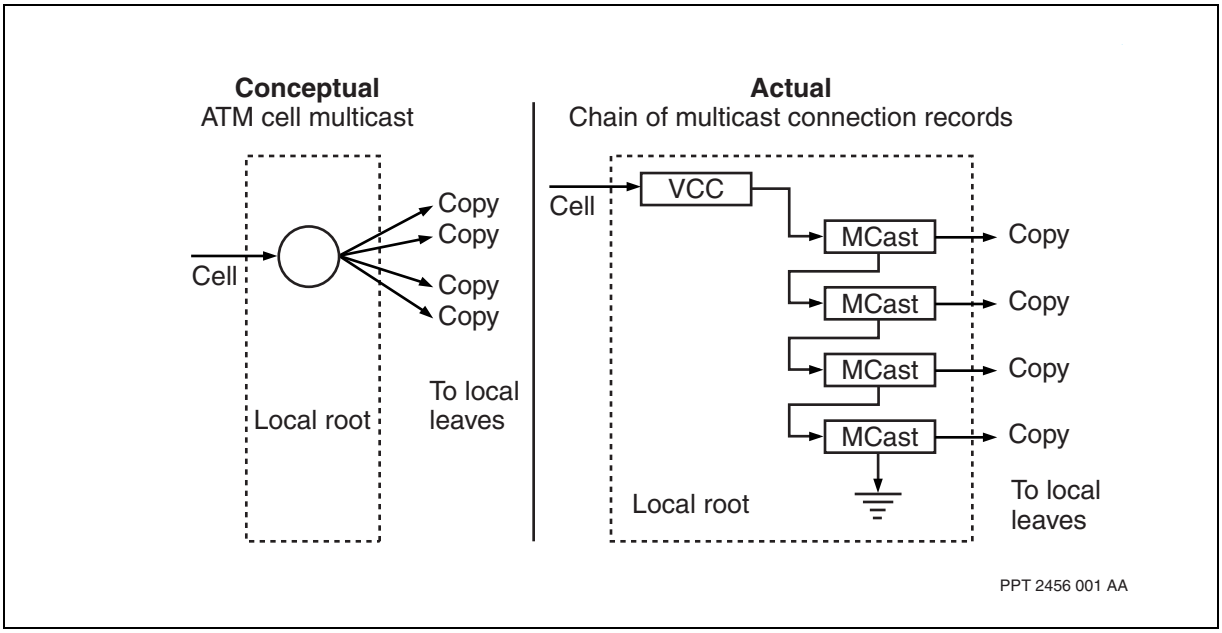
MSS 2455 001 AC

Multiservice Switch point-to-multipoint implementation

Nortel Multiservice Switch device's ATM function processor supports both spatial (to multiple ports) and logical (to multiple connections within a port) replication of cells. The figure ATM function processor support for cell multicast (page 182) shows how Multiservice Switch nodes use separate multicast connection resources for each copy of a cell.

Nodes implement ATM cell multicast as a variable length chain of connection records on the ATM function processor.

ATM function processor support for cell multicast

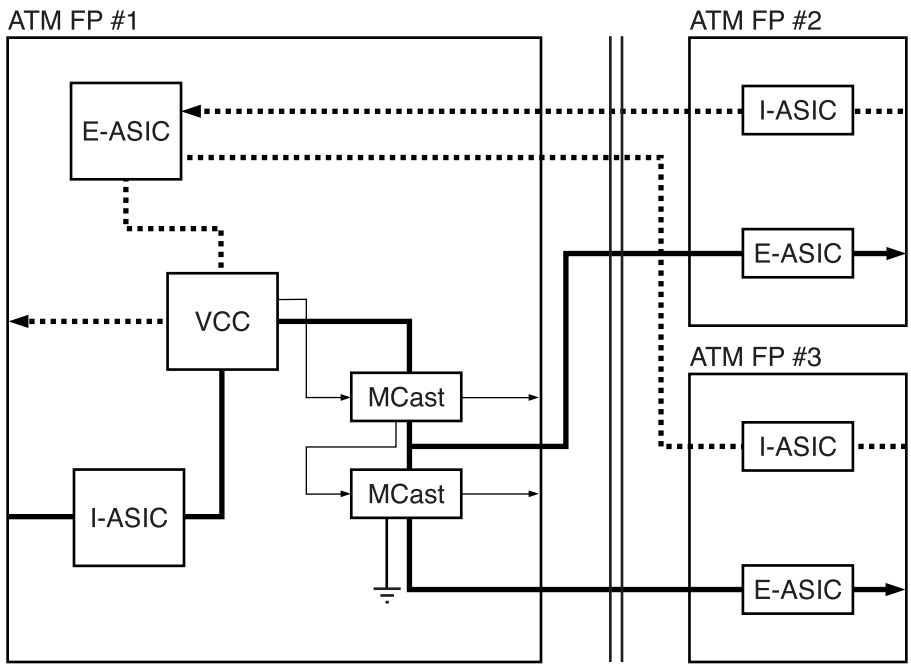


Each ATM function processor has two ASICs: one at the receive port and one at the transmit port. Multiservice Switch nodes perform the multicast function on the receive ASIC. The chain of connection records is a linked list that the node creates in software and manages internally through hardware. The multicast function lets cells flow in the forward direction, from the local root towards the local leaves. At the local root, the node replicates each cell once for each record in the chain of connection records, and sends the cell to the local leaf that the connection record identifies. The node repeats this function for all connection records in the chain.

The point-to-multipoint connection implementation supports reverse cell flows, from the local leaves to the local root. The Multiservice Switch node sends reverse cells directly to the VCC, bypassing the multicast mechanism. The network uses this cell flow only for OAM cells. Otherwise, point-to-multipoint connections are unidirectional. The figure ATM function processor cell flow (page 183) shows the forward and reverse cell flows.

For further details concerning point-to-multipoint OAM, see NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management*.

ATM function processor cell flow



Memory resources for point-to-multipoint connections

An ATM function processor has a limited amount of cell queue memory (CQM). CQM has two spaces:

- the connection space, which consists of connection context records that configure and monitor a particular connection
- the buffer space, which consists of cell and frame memory

Core connections refer to VCC and VPC connection resources. Multicast connection resources include

- a multicast connection type (VCC or VPC)

- a local connection index (LCI) for each record in the chain of multicast connection records

Nortel Multiservice Switch nodes support multicasting on the receive ASIC, as a result only the receive CQM has allocated space for multicast connections. The transmit CQM is distributed to the cell and frame memory according to the percentages specified in the *Lp Arc* components.

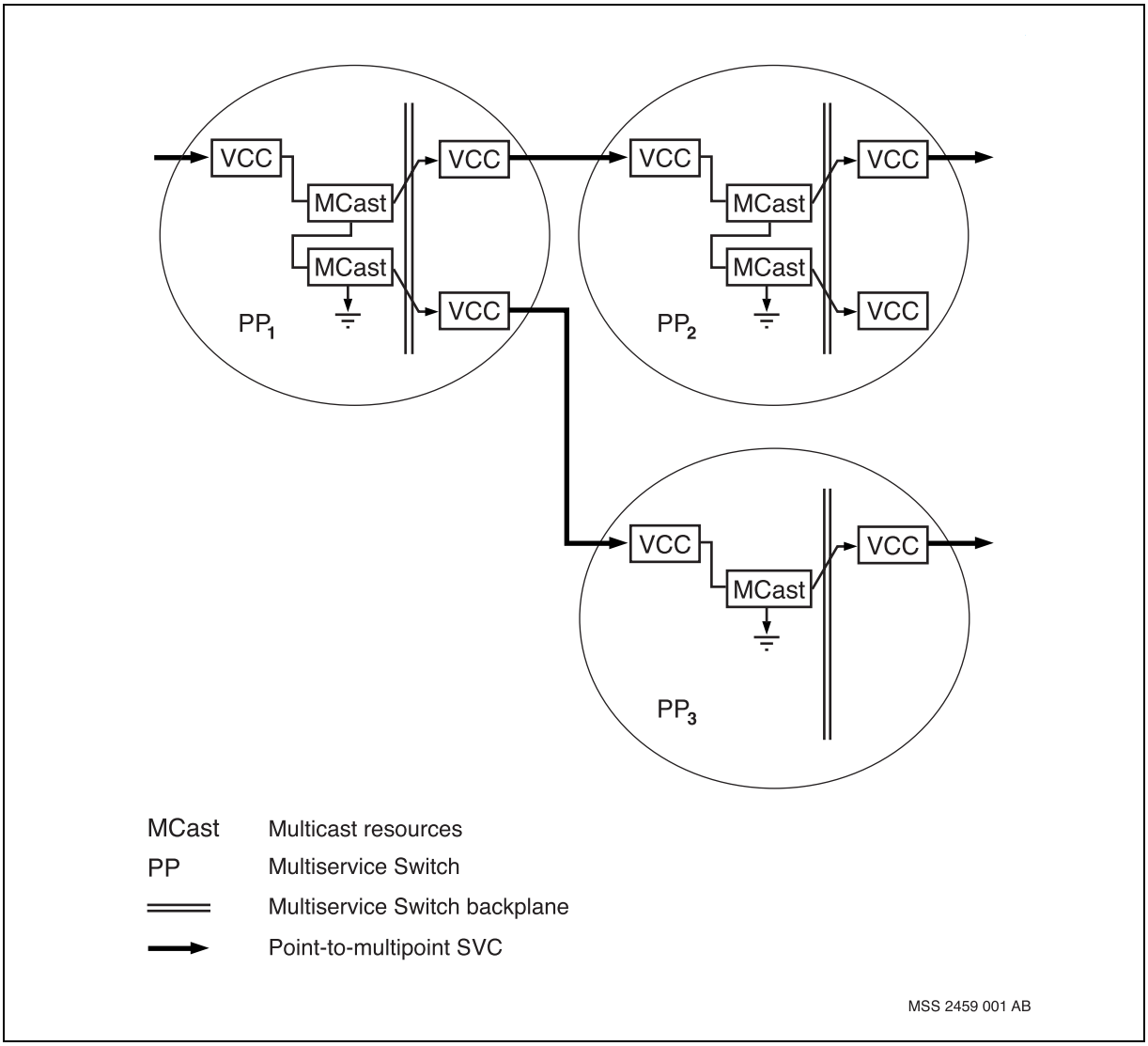
You configure connection and buffer space resources at the ATM function processor. Core and multicast connections are configurable. For more information on memory allocation, see the following documents:

- NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals*
- NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals*
- NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals*
- NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals*

Multicast resources

At each receive port in a point-to-multipoint connection, Nortel Multiservice Switch nodes use a multicast connection resource in addition to the point-to-point connection resources. A local root with *n* leaves uses *n* additional multicast connection resources. The figure ATM function processor multicast resources in the point-to-multipoint connection (page 185) shows the use of additional multicast resources within a typical point-to-multipoint connection.

ATM function processor multicast resources in the point-to-multipoint connection



Point-to-multipoint SVC and SPVC components

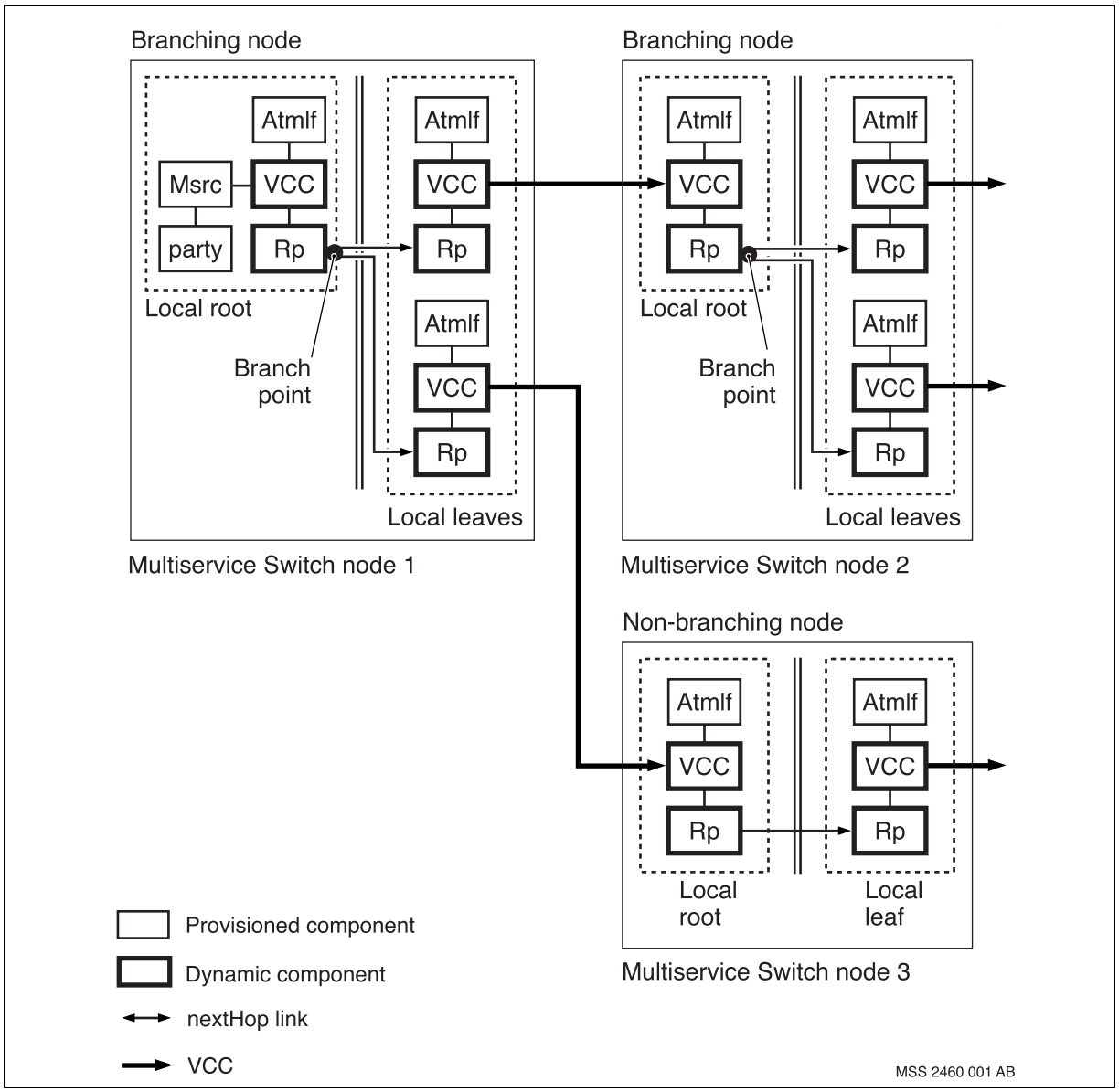
Point-to-multipoint SVCs and SPVCs use dynamic components. The figure Component hierarchy for point-to-multipoint SVCs and SPVCs (page 186) shows the relationship between components. Nortel Multiservice Switch nodes use relay points to dynamically connect multiple VCC components. Relay points link the local root across the backplane to the local leaves.

Multiservice Switch nodes associate each local root and local leaf with a single *Vcc* component (the same process that occurs for point-to-point SVCs and SPVCs). In addition, each local root has a unique attribute instance that identifies the relay point as the next hop for each associated local leaf. At the

branching node, the local root multicasts all information that it receives from the previous hop to all local leaves. Note that the multicast tree is unidirectional.

The *NextHop* attributes identify all the links to which the relay points are associated. ATM traffic management and OAM have operational attributes. Multiservice Switch nodes signal traffic and OAM configuration parameters through the UNI interface as part of the SETUP message.

Component hierarchy for point-to-multipoint SVCs and SPVCs



ATM interface types

Nortel Multiservice Switch nodes support point-to-multipoint SVCs and SPVCs on all ATM interfaces. For information on ATM interfaces, see NN10600-700 *Nortel Multiservice Switch 7400/15000/20000 ATM Technology Fundamentals*.

Addressing

Point-to-multipoint SVCs and SPVCs have no special addressing requirements. For information on addressing for dynamic networking, see the section ATM network addressing (page 35).

Signaling

Nortel Multiservice Switch nodes support point-to-multipoint SVCs and SPVCs on UNI, IISP, AINI, and PNNI interfaces. Call establishment, call clearing, and call error handling proceed, depending on the interface type. For more information on signaling, see the section ATM signaling (page 56).

Call routing

Nortel Multiservice Switch nodes use static routing or dynamic route discovery to forward call setup requests across the network. The routing protocol depends on the interface type. For more information on routing, see the section ATM routing (page 94).

Deployment

Point-to-multipoint SVCs and SPVCs support a node-by-node software upgrade for limited point-to-multipoint applications. Nortel Multiservice Switch nodes reject point-to-multipoint call setup requests at nodes that support only point-to-point connections.

Anycast point-to-point connections

The ATM anycast point-to-point capability allows an send station to connect to any member of a group at the far end using PNNI routing. The anycast capability is useful in network applications where an ATM address must represent a general service rather than a specific node or end station.

Navigation

Overview to anycast point-to-point connections

An end station requests an ATM anycast point-to-point connection through the setup message it sends across its user-network interface (UNI). The setup message for an anycast connection signals the following information:

- called party number in the called party number information element (CPN-IE), which defines the required ATM group address
- connection scope in the connection scope selection information element (CSS-IE), which limits the point-to-point connection request to group members within a specified level of the routing hierarchy

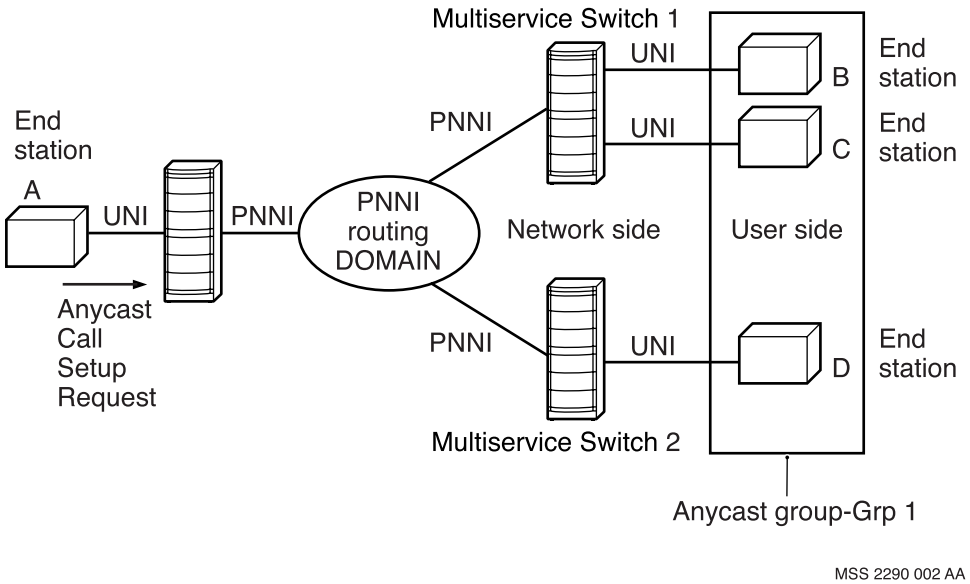
The edge node, upon receiving the setup message, establishes a point-to-point connection to one member of the group that is identified by the CPN-IE.

The figure ATM Anycast call requests (page 189) shows how an anycast call request can progress through a Nortel Multiservice Switch network. In this example, users B and C register the same group address, Grp 1, using integrated local management interface (ILMI) across the UNI connections to Multiservice Switch 1. In this way, users B and C declare that they are part of

the same anycast group. User D registers the same group address with the network side of its connection to Multiservice Switch 2. Users B, C and D are therefore all group members of anycast group Grp 1.

User A originates an anycast call request to group address Grp 1. User A does not know which group member will eventually terminate the point-to-point call. User A relies on the routing domain (PNNI in this example) to route the call to a node where an egress UNI interface declares reachability to group address Grp 1. At the terminating UNI interface, selection of a particular group member by Multiservice Switch 1 (User B or C) 1 is handled by the hop-by-hop routing mechanism. The connection scope in the incoming setup message determines which group member terminates the call.

ATM Anycast call requests



ATM group address format

The format of an ATM group address is the same as the format for an individual ATM endpoint address. The authority and format identifiers (AFI) for group address adhere to standards defined by the ATM Forum in *User-Network Interface Signalling Specification Version 4.0* (af-sig-0061.000). The table Allowable individual and group AFIs for ATM addresses (page 190) lists supported AFIs.

Allowable individual and group AFIs for ATM addresses

Individual AFI	Group AFI	Format
0x37	0xBB	X.121
0x39	0xBD	DCC
0x45	0xC3	E.164
0x47	0xC5	ICD
All values are in hexadecimal.		

Characteristics and scope of group membership

An ATM group and its members have the following characteristics:

- a group address identifies an ATM group
- an ATM group represents a collection of ATM end systems
- an ATM group has one or more members, where each member is an ATM end system
- an ATM end system can be a member of any number of ATM groups, or may not be a member of any group
- an ATM end system may join or leave a group using ILMI dynamic address registration and deregistration procedures

Each ATM group address is associated with membership scope which specifies the inclusive routing hierarchy in which the member is known. There are 15 levels of organizational scope hierarchy for ATM group addresses. The table Levels of organizational scope hierarchy (page 190) summarizes these levels.

Levels of organizational scope hierarchy

Level number	Label	Description of scope
1		This label maps to a physical network. For example, the mapping can be to a bottom level peer group, or to a peer group of higher level in the PNNI routing hierarchy to simulate extended physical networks.
2 3 4		These labels map to ATM sub-networks that do not use inter-building or wide-area links.
		(1 of 2)

Levels of organizational scope hierarchy (continued)

Level number	Label	Description of scope
5		This label maps to the inclusive routing hierarchy that is not geographically separated. This mapping allows the network operator to confine the traffic within a local location and avoid using wide-area links or inter-building links.
6 7		These labels identify ATM networks that may use inter-building links or wide-area links.
8		This label identifies ATM networks that represent the inclusive routing hierarchy of an autonomous organization. An autonomous organization is an organization that has administrative authority over the network. The ATM networks identified by this membership scope can use inter-building and wide-area links.
9 10		These labels identify the union of more than one organizations.
11 12 13 14		These labels identify ATM networks that represent a collection of autonomous organizations that are organized by a provider or organizational partnership.
15		This label represents all autonomous organizations which form a connected private ATM network.
Level 15, global, represents the widest scope.		
(2 of 2)		

The UNI membership scope is mapped to a PNNI routing level during routing. See ATM group addresses and PNNI (page 191) for more information on the default UNI to PNNI scope mapping.

ATM group addresses and PNNI

Nortel Multiservice Switch maps both the configured UNI group address membership scope and the connection scope defined in the CSS-IE into the PNNI routing level. This mapping approach allows flexibility in restructuring the routing levels within a network without affecting the membership scope control for an end station.

The UNI to PNNI mapping is configurable under ATM routing for PNNI. The table Default UNI scope to PNNI level mappings (page 192) summarizes the default mappings as they are defined in *Private Network-Network Interface (PNNI) Specification Version 1.0* (af-pnni-0055.000).

Default UNI scope to PNNI level mappings

UNI scope	PNNI routing level indicator
1-3	96
4-5	80
6-7	72
8-10	64
11-12	48
13-14	32
15 (global)	0

Group address registration

To support group address registration, signaling relies on current ILMI dynamic address registration and de registration procedures. The ATM Forum specification for UNI 4.0 extends the ILMI procedure to support the ATM address organizational scope management information base (MIB) object.

The service provider configures the scope of an ATM group address on the user side of a UNI. To configure an ATM group address, a group address AFI is configured on the network side. See the table Allowable individual and group AFIs for ATM addresses (page 190) for permitted AFIs.

Through ILMI dynamic address registration, every end system identifier (ESI) on the user side of a UNI is concatenated with every group prefix on the network side. This concatenation forms the set of ATM group addresses. Nodes register these group addresses on the network side of the UNI. The membership scope is stored together with the actual group address, and both are registered with the call router.

A Nortel Multiservice Switch node that operates as the user side of a UNI does not support configuration of entire dynamic ATM group addresses. However, a Multiservice Switch node that operates as the network side of a UNI, connected to a third party vendor switch, accepts registration of group addresses which are entirely provisioned on the user side of the UNI.

Signaling and information elements for ATM anycast

The CSS-IE in the setup message allows a calling user to constrain a point-to-point connection request to group members within a specific level of routing hierarchy.

The connection scope selection information element (CCS-IE) signals the connection scope for an ATM anycast call request. When a node receives a message that includes a CSS-IE, the call progresses with the CSS-IE unchanged. If the CSS-IE is absent from a setup message that contains a group address in the CPN-IE, the network assumes a UNI membership scope of localNetwork.

Routing through PNNI

Anycast call requests use the PNNI routing mechanisms to select a member within the specified ATM group.

- PNNI routing example for anycast connections (page 193)
- PNNI routing considerations (page 195)

PNNI routing example for anycast connections

The UNI to PNNI scope mapping supports connection requests under two circumstances:

- to determine reachability information for a group address in a PNNI routing domain
- to process anycast connection requests with a group address defined in the CPN-IE

When determining reachability information for a group address in a PNNI routing domain, the node maps the membership scope of a configured group address to a PNNI routing level. This mapping allows advertisement of the reachability of group addresses according to the standards-based address scoping rules of PNNI. Reachability information depends on the routing hierarchy level of the peer groups in the routing domain.

As an example of advertising the reachability of group addresses into higher level peer groups, see the example in the figure Example of group address advertisement in a PNNI hierarchy (page 195). Assume that node A26 advertises reachability to the following group addresses:

- G0 (maps to PNNI level 104)
- G1 (maps to PNNI level 96)
- G2 (maps to PNNI level 80)
- G3 (maps to PNNI level 70)

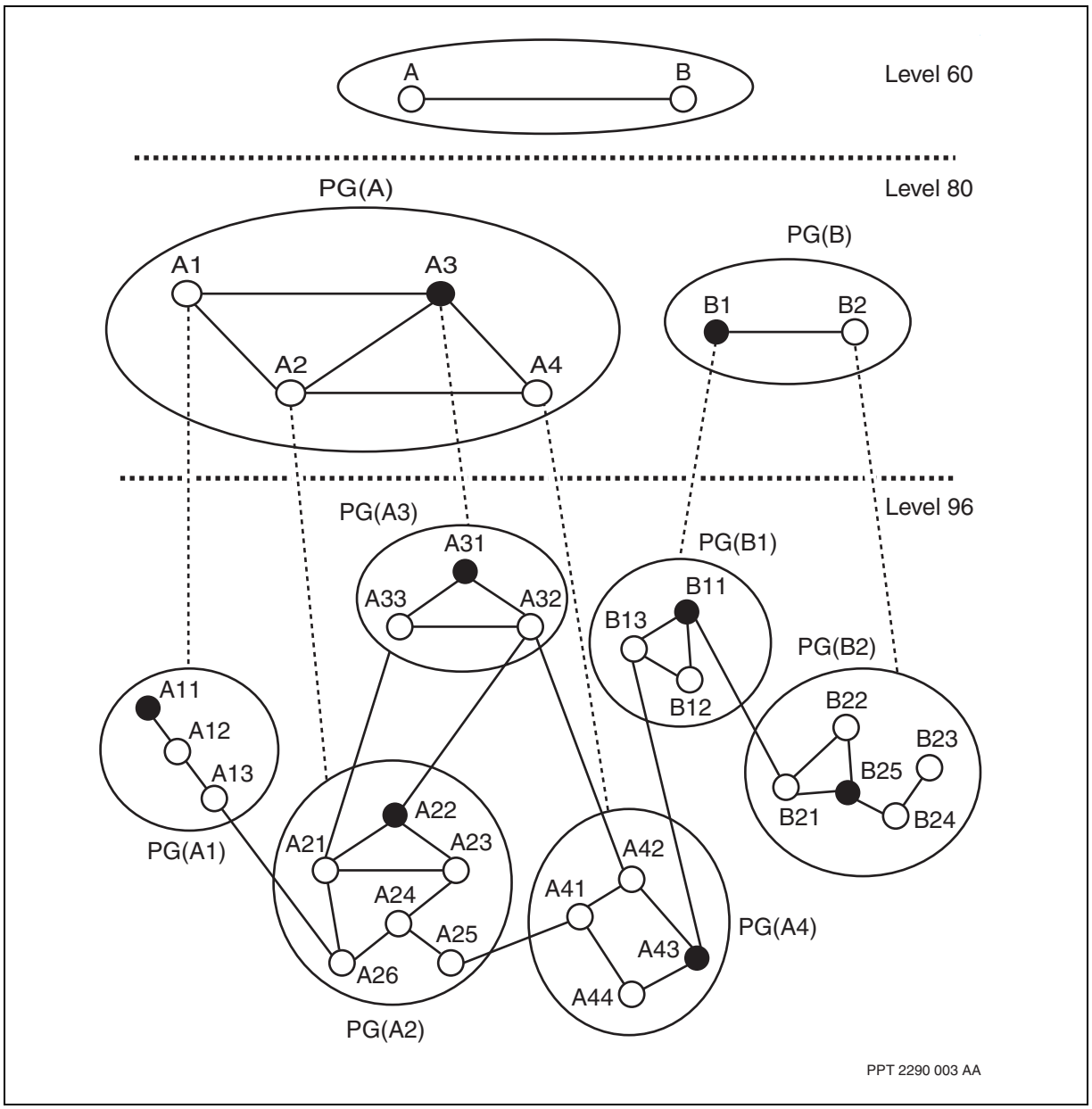
However, since node A26 is part of a peer group at routing level 96, it can advertise to higher levels in the hierarchy only group addresses G1, G2, and G3. Since the membership scope of address G0 is lower than the level of its peer group, node A26 cannot advertise that group address. Logical group node (LGN) A2 receives the advertised addresses from A26, and further refines its scope of advertisement. LGN A2 advertises group addresses G2

and G3 only, since the scope of G1 is lower than the scope of its peer group. Finally, LGN A receives the advertised addresses from LGN A2, and further refines its scope of advertisement. LGN A does not advertise any of these group addresses, since the scope of both addresses is lower than the level of its peer group.

In the case of identical group addresses reachable within the same peer group, reachability to the address with the higher of the two scopes is advertised.

For connection requests in an anycast call request, the signaled membership scope in the CSS-IE is mapped to a PNNI routing level. This mapping allows the network to select a route to a group member within the specified connection scope. In other words, when requesting a route for an anycast call request, the node supplies a connection scope (mapped to a routing level). This scenario assumes that the CPN-IE in the setup message defines a group address.

Example of group address advertisement in a PNNI hierarchy



PNNI routing considerations

For an anycast call request, the signaled scope in the CSS-IE specifies the scope of the connection. The routing system cannot route the call through any peer group that has a routing level that is higher than the specified connection scope. For example, consider an anycast call request with a connection scope of intraSitePlusOne. This connection scope maps to a PNNI routing level of 72. That is, the network routes the call only through peer groups of level 72 or lower.

In the case of identical group addresses reachable from the same peer group but with different scopes, the routing level of the top-level peer group to which the calling user belongs is inferred. Inference is completed through the top designated transit list information element (DTL-IE) in the designated transit list (DTL) stack. (The PNNI protocol uses the DTL-IE to set up routes.) With this information, the router can determine which reachable group address to select.

As an example, assume node A11 in the figure Example of group address advertisement in a PNNI hierarchy (page 195) wants to set up an anycast call to group address G1, which maps to PNNI level 96 as defined in the section PNNI routing example for anycast connections (page 193). This address is reachable through LGN A2. See the example in PNNI routing example for anycast connections (page 193). That is, its reachability is known to A11 through uplink (A11-A2). A connection scope of 79 or lower results in the call failing to route to the desired destination. The connection scope must be 80 or higher for the call to successfully route to A26.

As another example, assume the following:

- node B21 advertises reachability to address G1 with scope 60
- node B25 advertises reachability to G1 but with scope 80

Through the rules of address scoping, LGN B advertises reachability to address G1 since at least one of the addresses in peer group (PG) B2 has a scope of 60. Assume that node A11 wants to setup a call request to anycast address G1 and that the connection scope is set to be 60. By the time this call finds its way to PG(B2), the highest level DTL in the stack is:

DTL: [A,B] pointer-2

From this DTL, the entry border node in the terminating peer group, B21, infers that the call originated from PG(A), and must terminate in PG(B). Therefore, for a call request to G1 to have made it this far, and originating from peer group A, the connection scope in the CSS-IE must have been 60 (or higher). The entry border node then knows to expand the DTL in such a way that the address G1 at node B21 is chosen for terminating the call. The DTL stack therefore carries enough information in it for entry border nodes to decide which anycast address to select as a destination, in the case of identical addresses reachable from the same peer group.

If the CSS-IE is missing from an anycast call request, the node assumes a connection scope of localNetwork (PNNI routing level 96).

Networking scenarios

The private network-to-network interface (PNNI) nodes exchange the reachability information that they acquire from the user-to-network interfaces (UNIs), the interim inter-switch protocol (IISPs) interfaces, the ATM inter-network interfaces (AINIs) and the PNNIs. Nortel Multiservice Switch nodes use this information to route calls across the backbone. These configurations support

- point-to-point switched permanent virtual connections (SPVC)
- point-to-point switched virtual connections (SVC)
- point-to-multipoint SPVCs
- point-to-multipoint SVCs
- point-to-point switched permanent virtual paths (SPVP)
- point-to-point switched virtual paths (SVP)

Navigation

- Permanent paths that support virtual interfaces (page 197)
- Hybrid network (UNI, IISP, AINI, and PNNI) (page 199)
- Multi-vendor single peer group PNNI network (page 200)
- Multi-vendor multiple peer group PNNI network (page 202)

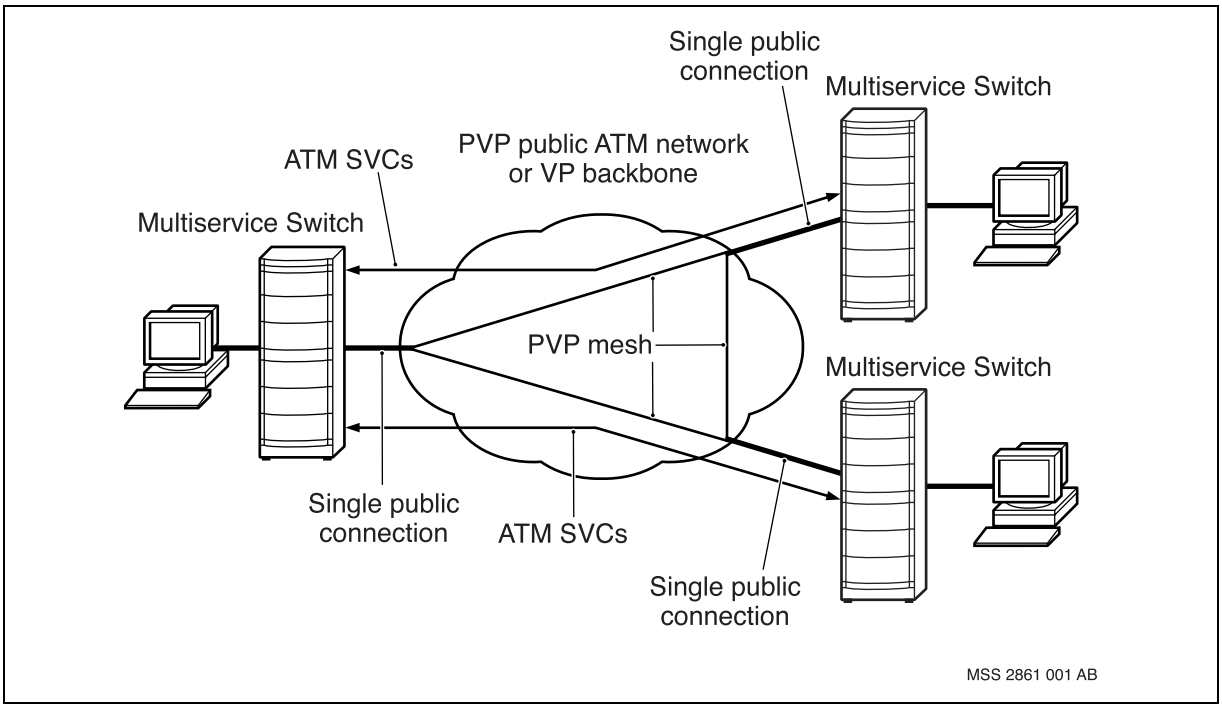
Permanent paths that support virtual interfaces

This set of scenarios show typical applications for tunnelling SVCs over permanent virtual paths (PVP).

The figure SVC tunnelling through a PVP network (page 198) shows how three Nortel Multiservice Switch nodes connect through a PVP mesh. The mesh can be either a public network or a virtual path backbone.

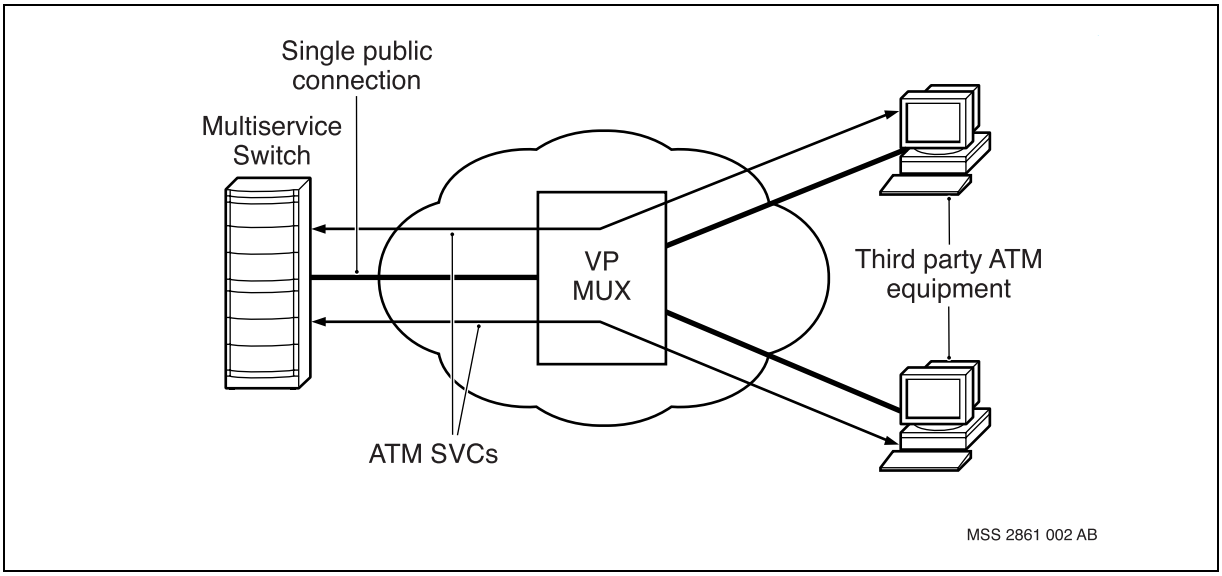
The configuration on each node must include correct mapping of the local VPIs to the VCIs that the SVCs use to identify the connection from end to end. There are no configuration requirements for the mesh other than to ensure that the PVPs are correctly set up.

SVC tunnelling through a PVP network



The figure SVCs through a virtual path multiplexer (page 199) shows how a Multiservice Switch node can connect to third-party ATM equipment through a virtual path multiplexer. In this example, a single PVP tunnels SVC to multiple ATM end points. The configuration requirements include correct mapping of the local VPI to the VPCI on the node and the multiplexer. The multiplexer handles call setup to the third-party end points.

SVCs through a virtual path multiplexer



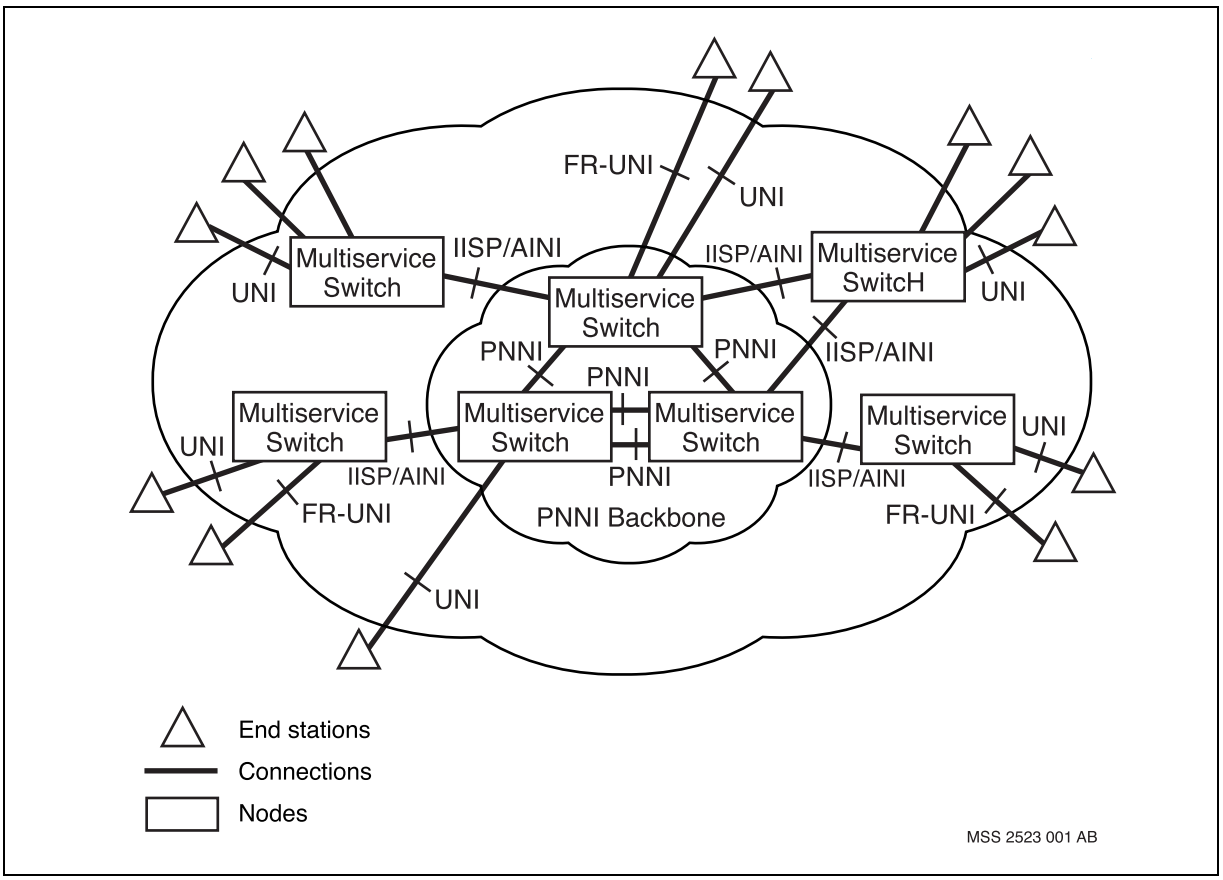
Hybrid network (UNI, IISP, AINI, and PNNI)

This networking scenario involves Nortel Multiservice Switch nodes in an IISP/AINI-based network with a PNNI-based backbone.

This network consists of a single-level PNNI network which serves as a backbone to an IISP/AINI-based network. The PNNI backbone provides the core dynamic routing function. This scenario shows how an existing Multiservice Switch IISP/AINI network can evolve to PNNI.

The figure Hybrid networking scenario (UNI, IISP, AINI and PNNI) (page 200) illustrates this scenario.

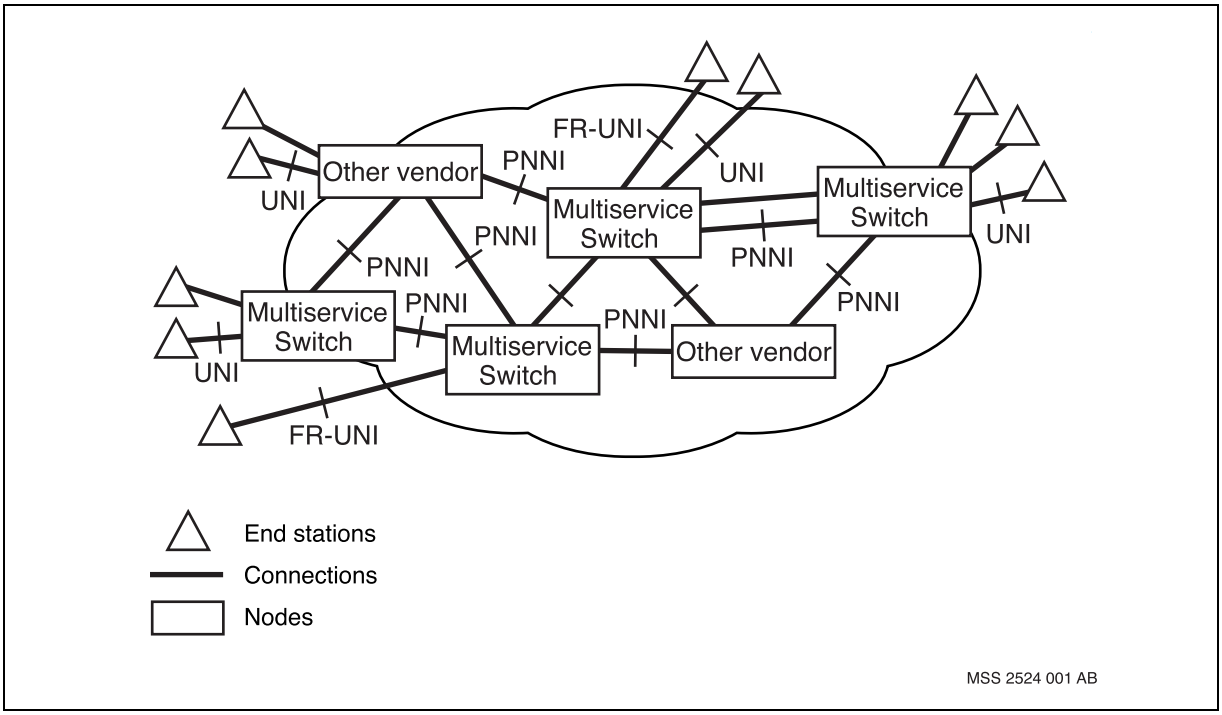
Hybrid networking scenario (UNI, IISP, AINI and PNNI)



Multi-vendor single peer group PNNI network

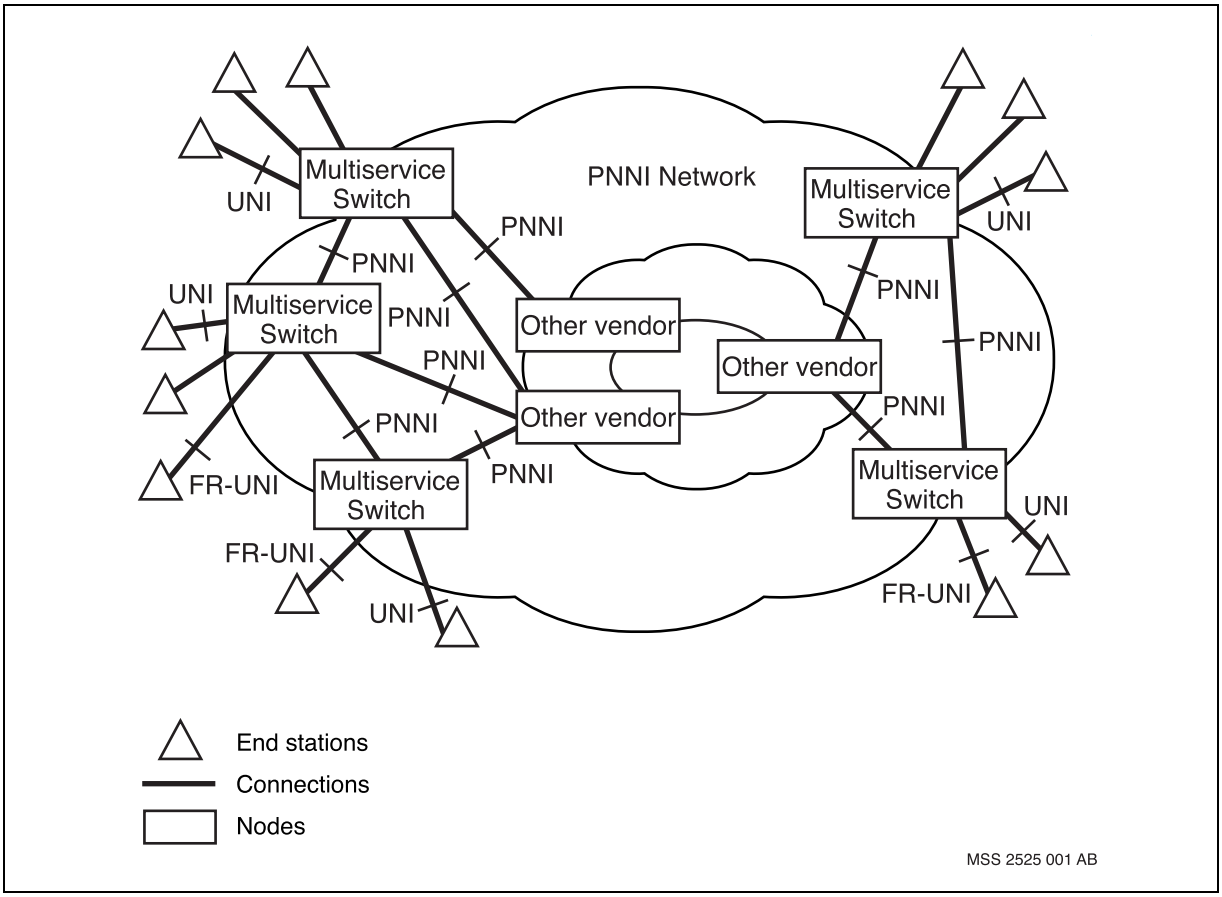
Nortel Multiservice Switch ATM networking lets you build single-level PNNI networks, either with Multiservice Switch nodes exclusively, or with devices from other vendors. The figure Multi-vendor single peer group PNNI network - scenario 1 (page 201) illustrates this mixed-vendor scenario.

Multi-vendor single peer group PNNI network - scenario 1



The figure Multi-vendor single peer group PNNI network - scenario 2 (page 202) illustrates another networking scenario that connects multiple sites into a single peer group PNNI network. This topology requires setting up routing control channels (RCCs) over permanent or switched virtual paths. Because Multiservice Switch nodes do not support VP-associated signaling, this configuration includes non-Multiservice Switch equipment which does provide this support.

Multi-vendor single peer group PNNI network - scenario 2



Multi-vendor multiple peer group PNNI network

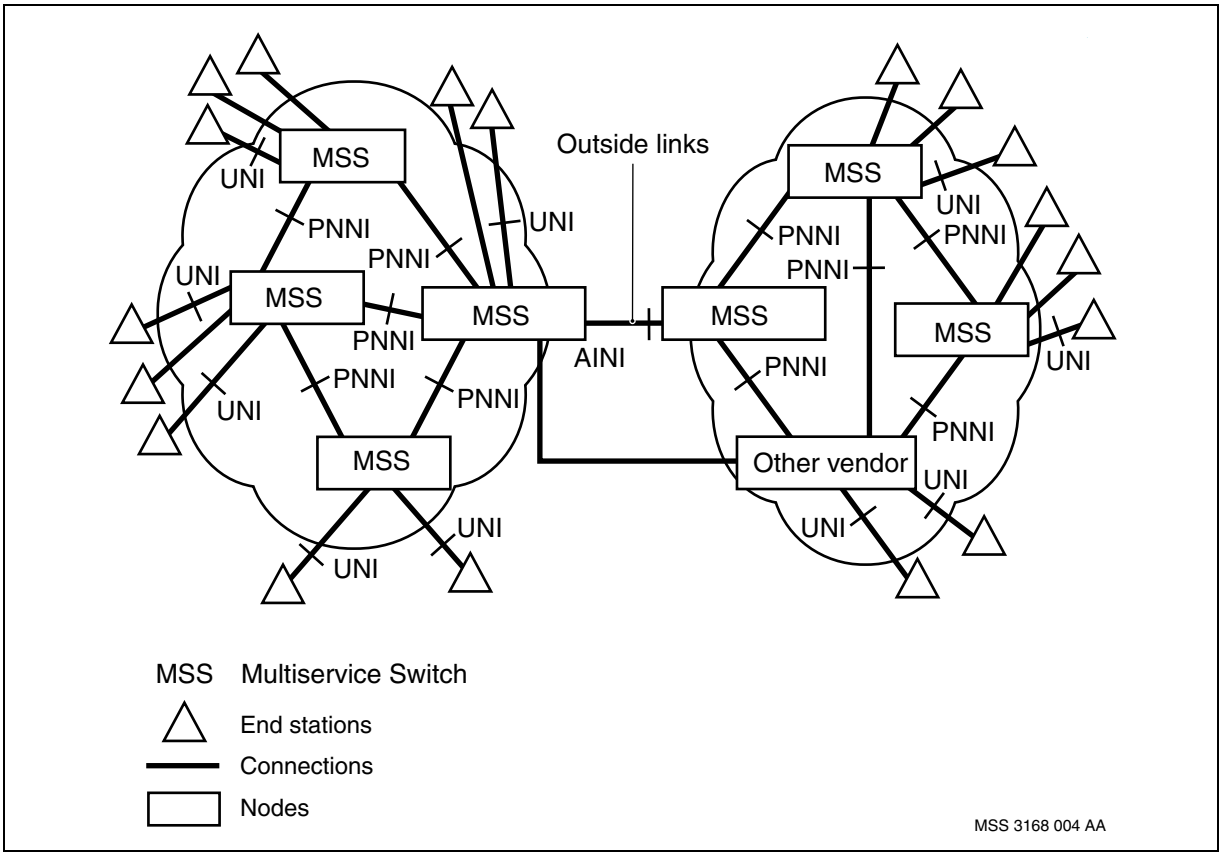
Nortel Multiservice Switch nodes can assume any role in a peer group and hence, can fully participate in a multiple peer group (hierarchical) PNNI network:

- act as a border node
- act as a peer group leader (PGL) at any level in the hierarchy
- represent the peer group as a logical group node (LGN) at the higher level
- act as a peer to the LGN at the lowest level

A node that has physical links to another node that belongs to another peer group is a border node. A PGL presents an aggregated view of its peer group to PGLs elected in other adjacent peer groups at the same level.

The figure Multi-vendor multiple peer group PNNI network scenario (page 203) illustrates this networking scenario.

Multi-vendor multiple peer group PNNI network scenario



Connection mapping

For APC-based function processors and 2- and 3-port CQC-based function processors, you define a usable range within the connection map for each ATM interface. You define this range for each ATM interface on each function processor for all Nortel Multiservice Switch nodes in the network.

Eight-port CQC-based function processors have the same characteristics as ATM IP function processors with the exception that they do not support VPI number 4095. This appendix does not provide information on 8-port ATM function processors, since these function processors do not require configuration through the *ConnectionMapping* component. See Connection map address assignment for ATM IP function processors (page 57) for more information on connection mapping for 8-port CQC-based function processors. See also Considerations for node migration to Release 6.0 (page 225) for information on migration between releases.

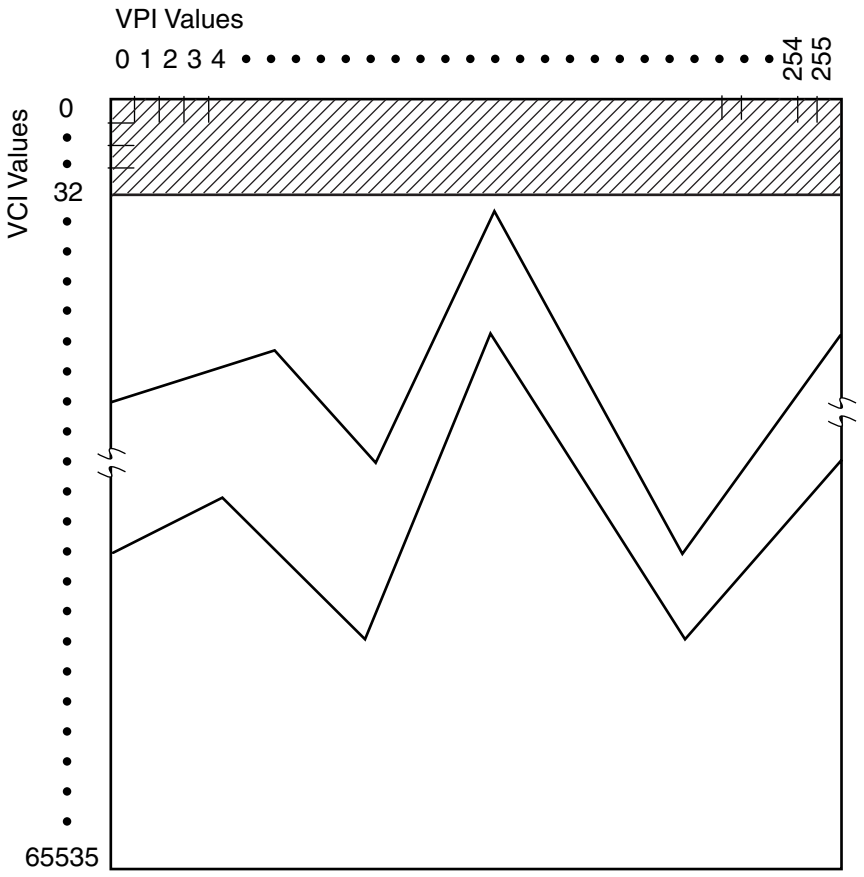
Navigation

Scope of the connection map space

For the 2- and 3-port CQC-based function processors, the connection map is a theoretical range of possible address spaces based on a two-axis grid of 256 virtual path identifiers (VPI) and 16128 virtual channel identifiers (VCI). This space represents the maximum range of possible addresses from which the service provider defines the usable range of addresses in the connection map. That is, the usable range within the connection map is the range of positions that are available to connections under the interface and is a subset of the overall domain.

The figure Connection map range of possible addresses (2- and 3-port CQC-based function processors) (page 205) shows the theoretical range of possible addresses.

Connection map range of possible addresses (2- and 3-port CQC-based function processors)



Note: The VCI range of 0-31 is reserved for use by future standards.

PPT 3094 002 AA

For the APC-based function processors, the connection map is a theoretical range of possible address spaces based on a two-axis grid of

- 256 virtual path identifiers (VPI) and 16384 virtual channel identifiers (VCI) for UNI applications
- 4 096 virtual path identifiers (VPI) and 16384 virtual channel identifiers (VCI) for NNI applications

This space represents the maximum range of possible addresses from which the service provider defines the usable range of addresses in the connection map. That is, the usable range within the connection map is the range of positions that are available to connections under the interface and is a subset of the overall domain.

The usable range in the connection map space consists of two, one, or no VCC spaces, and a VPC space:

- a VCC space under VPI=0 (optional)
- a VCC space under a user-defined set of one or more non-zero VPIs (optional)
- a VPC space defined by the VPIs that are not used for the VCC spaces (mandatory)

You define a usable range for each ATM interface in the network. Each interface can have a different or the same usable range as other interfaces in the network, depending on the following conditions:

- the number of connections under the interface
- addressing requirements of equipment at the other end of each connection
- network design and VPI.VCI numbering plan considerations

The connection map configuration for a particular interface does not affect either the number of available connections or the usable configurations for other interfaces.

The most common and the simplest approach is to define the same connection map for all ATM interfaces on the APC-based function processors and 2- and 3-port CQC-based function processors across the Nortel Multiservice Switch network. However, network complexity and the immediate surroundings of a node can determine different connection maps for two or more interfaces on that node.

Considerations for planning the connection map

When you plan the connection map for an ATM interface, there are four important considerations for resource and network management:

- the number of connections that the physical port supports
- the internal numbering plan for VPIs and VCIs in the Nortel Multiservice Switch network
- the numbering plan requirements of third-party equipment at the far end of each connection that the interface supports

- the required number of VCCs and VPCs on the interface

Proper planning reduces or eliminates oversights in resource allocation and the need to change attribute values and manually reconfigure the ATM connections. When planning is complete, you can proceed to configure the usable range in the connection map.

For more information on planning the connection map, see the following sections:

- Objectives of the planning process (page 207)
- Overview of the planning process (page 208)
- Requirements for planning information for 2- and 3- port DS1/E1 function processors (page 209)

Objectives of the planning process

When you plan the usable range in the connection map space, you are defining both an address space and resource usage in a real-time networking environment. You need to do the following:

- Decide how to best use function processor resources. The more connections a function processor supports the less memory is available for frame and cell buffers.
- Determine the internal and external numbering plan requirements of the network.
- Define a usable range in the connection map space that serves both current and future numbering plan requirements and the network design standards that engineering has established for the internal network. The usable range is a balance between the range of connection addresses available to the interface and function processor memory usage.
- Determine the maximum number of VCCs and VPCs that the interface must support. This number is a balance between the number of connections that the interface supports and function processor memory usage.
- Determine the values needed to configure the attributes that govern resource usage, the usable connection map, and interface maximum connection requirements through these key attributes:
  - *connectionPoolCapacity* attribute under the *Arc Override* component or under the *Arc Cqc Override* component
  - *connectionPoolCapacity* attribute under the *Arc Override* component or under the *Arc Apc Override* component
  - *numVccsForVpiZero* (*nZVccs*) attribute under the *ConnectionMapping Override* component

— *numNonZeroVpisForVccs* (*nVpis*) attribute under the *ConnectionMapping Override* component

— *numVccsPerNonZeroVpi* (*nVccs*) attribute under the *ConnectionMapping Override* component

— *firstNonZeroVpiForVccs* (*firstVpi*) attribute under the *ConnectionMapping Override* component

— *maxVccs* attribute under the *AtmInterface* component

— *maxVpcs* attribute under the *AtmInterface* component

For APC-based function processors, you can use either the *connectionPoolCapacity* attribute under the *Arc Override* component or the *connectionPoolCapacity* attribute under the *Arc Apc Override* component.

You must use the *connectionPoolCapacity* attribute under the *Arc Override* component to define the maximum number of connections available on an LP bound to either an 8-port CQC DS1 ATM function processor or an 8-port CQC E1 ATM function processor.

For the 2- and 3-port CQC-based function processors, you can use either the *connectionPoolCapacity* attribute under the *Arc Override* component or the *connectionPoolCapacity* attribute under the *Arc Cqc Override* component.

The following rules apply when defining *connectionPoolCapacity*:

- For eight-port DS1/E1 ATM function processors, the value of *connectionPoolCapacity* must exceed the sum of VCCs, VPCs and VPTs on all ATM interfaces using the LP. This sum includes all ATM interfaces served by either independent ATM links and IMA link groups. In this case, there is no dependence on the VCC space defined by the connection map.
- For all other CQC-based function processors, the value of *connectionPoolCapacity* is divided evenly among all ports available. The number of connections supported on each port must exceed the range specified in the connection map for each ATM interface bound to a port on the LP.

Overview of the planning process

Follow these general steps when planning the connection map:

1 Determine how many connections the port associated with the interface must support.

2 Assess the addressing requirements for Nortel Multiservice Switch nodes and third-party equipment, both internal and external to the Multiservice Switch network. As far as possible, take future requirements into consideration.

3 Consider internal network standards for addressing.

4 Define a numbering plan that accommodates these addressing requirements.

5 For the interface, determine if a VCC address space under VPI=0 is needed. If needed, determine the range of VCC address positions that are needed under VPI=0.

6 For the interface, determine if a VCC address space is needed under a set of contiguous non-zero VPIs. If needed, follow these optional steps:

7 Determine how many non-zero VPIs are needed to accommodate the range of VCC address positions in this space.

8 Determine the range of VCC addresses that are needed under non-zero VPIs. This range is the same for all VPIs in this space.

9 Determine the value of the first VPI in the range of contiguous non-zero VPIs.

10 Determine the maximum number of VCCs and VPCs that can be supported under the ATM interface.

11 Translate these requirements into values that can be configured for the connection map and ATM interface.

Requirements for planning information for 2- and 3- port DS1/E1 function processors

The requirements for the relationship between the information derived through the planning process can be expressed for all 2- and 3-port DS1/E1 ATM function processors as follows:

<number of connections per port> ≥ <connection map range>

The *Arc* subcomponent of the *Lp* component defines the <total number of connections> and <number of connections per port>.

In addition to the interaction between the values derived through each of these steps, there are a number of considerations and constraints. The following section provides an example to illustrate how to set up a connection map.

Planning process example for CQC-based function processors

This example is based on a connection map with the following characteristics:

- 1536 connections to be supported by the port
- a range of 768 usable VCC address positions under VPI=0
- three non-zero VPIs for usable VCC address positions under non-zero VPIs
- non-zero VPIs are 1, 2, and 3

- a range of 64 VCC address positions for each non-zero VPI for a total of 192 VCCs under non-zero VPIs (256 including those under VPI=0)
- 959 maximum VCC address positions and 252 maximum VPC address positions within the usable range defined by the connection map

The example assumes the following about the network:

- The interface is part of a self-contained Nortel Multiservice Switch ATM network with no unusual or external addressing requirements at this time but with possible connections to external equipment in the future.
- The internal network standards specify that addresses are assigned by department group.
- There are three groups: finance, sales, and purchasing. There is also a central administration.
- All connections are permanent virtual connections (PVC) at this time, but switched virtual connections (SVC) are planned for the near future.

This example describes the planning steps and associated considerations and constraints. See Further planning considerations for CQC based function processors (page 224) for additional general considerations.

The table Planning example for configuring the ATM connection map for CQC-based function processors (page 211) presents the example. The figure ATM interface connection map model for CQC-based function processors: walk-through example (page 223) shows a model representing the connection map in this example. These parameters are for demonstration purposes and are part of an arbitrary selection that does not take into consideration external networks and switches that are customer network specific.

The figure ATM interface connection map model for CQC-based function processors: walk-through example (page 223) shows the connection map for this example. Compare this example to the default connection map in the figure ATM interface connection map for CQC-based function processors: default values (page 224).

Planning example for configuring the ATM connection map for CQC-based function processors

	Step	Considerations and constraints
1	Determine how many connections the port associated with the interface must support. This information is for configuring the *connectionPoolCapacity* attribute under the *Arc Cqc Override* component associated with the logical processor. **Syntax** The value of the connection pool capacity must be • between 512 and 4096 for ports 0 and 1, and 512 and 2560 for port 2 • evenly divisible by 256 Some function processors have only two ports. In this case, the connection pool capacity for non-existent ports must be set to zero.	The number of connections supported on the port is the connection pool capacity. As the number of supported connections increases, the amount of CQM used to support connections increases, leaving less memory available for queues and buffers. That is, you must balance the number of connections supported on the port against the amount of memory available for queues and buffers to ensure that ATM service category expectations are met. Good resource management helps ensure that the node does not discard cells or frames while the configuration settings unnecessarily reserve memory to support connection spaces that the network rarely or never uses. See resource management topics in the following documents: • NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals* • NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals* • NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals* • NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals* The connection pool capacity must be greater than or equal to the total number of available VPC/VCC address positions in the connection map. This number includes 256 connections for VPCs, regardless of whether the network uses all VPCs. Further, connection pool capacity also includes all VCC space in the connection map, rounded up to the next multiple of 256.

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
1	(continued)	The total number of VCC address positions includes the range under VPI=0 and the range under non-zero VPIs. The total number of VPC address positions defined for the connection map is defined as either • 256 if the number of VCC positions under VPI=0 is zero (no VCC positions in the connection map), or • 256 - <# of non-zero VPIs> - 1 if the number of VCCs for VPI=0 is greater than zero. For the network example in the walk-through, set this value to 1536 (the attribute default).
2	Assess the addressing requirements for Multiservice Switch and third-party equipment. The addressing requirements define how the usable connection map space is set up.	All addressing requirements need to be noted for equipment that is assessable to the nodes in the network. Multiservice Switch can use all VPI.VCI address positions in the overall connection map domain with the exception of VPI.VCI/0.0, which is reserved for an idle position according to ATM Forum standards. VPI.VCI/0.1 and VPI.VCI/0.2 are restricted. VPI.VCI/0.2 to VPI.VCI/0.32 is reserved for future use by ATM standards.If only Multiservice Switch equipment requires addressing through the network, the addressing requirements are simplified, limited only by internal network standards. Some third-party equipment can have specific addressing requirements, such as support for path VPI/0 or specific VCI ranges under non-zero VPIs. These specific requirements can introduce greater complexity when defining address requirements.

(2 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
3	Consider internal network standards for addressing.	From the perspective of setting up the usable connection map, the service provider's internal networking standards define how addressing positions are allocated to subscribers. In the example network for this walk-through, the positions under VPI=0 can be allocated to administration, while the positions under non-zero VPIs can be allocated one each to finance, sales, and purchasing, with one reserved for future use.
4	Define a numbering plan that accommodates these addressing requirements.	Complete this activity at the network level. The numbering plan includes information from steps 2 and 3, and defines the addressing requirements that the usable connection map for each interface must meet. Specifically, it defines • how many address positions are needed under VPI=0 • how many address positions are needed under non-zero VPIs • specific address requirements

(3 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
5	For the interface, determine if a VCC address space is needed under VPI=0. If needed, determine how many address positions are required. This information is for configuring the *numVccsForVpiZero* (*nZVccs*) attribute under the *ConnectionMapping Override* component. **Syntax** The value defining the range of VPI=0 address positions follows these rules: • The number of positions in the range can be 0. • If not 0, the number of positions must be a multiple of 256 to a maximum of 16 128. Typical configurations set this value between 256 and 2048.	If the interface needs a VCC address space, use the space under VPI=0. If the interface must support a connection to equipment that uses VPI/0, the value for the attribute is 0 (zero) and no VCC address positions are permitted under VPI=0. In this case, the interface supports only VPCs. Further, if VCCs must be relayed through or terminated at this Multiservice Switch node, configure these VCCs under another interface. This value identifies the number of positions in the range, where the range always starts at VCI=0. However, VPI.VCI/0.0-31 is reserved and should not be used as a configurable address for a VCC. This reserved position is a further restriction on the usable connection map address space. For example, if you set this value at 256, the resulting range of available address positions is VPI.VCI/0.32 to VPI.VCI/0.256—that is, 224 positions. The range of available VCI values under VPI=0 (represented as V) is defined as V = *nZVccs* - 32 which accounts for the reserved position. Internal network numbering standards can also define a set of address positions to be reserved for future standards or applications. In Multiservice Switch, VPI.VCI/0.1 to VPI.VCI/0.31 are typically reserved for future standards.

(4 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
5	(continued)	Observe the following requirements for SVCs: • The network side automatically selects address spaces under VPI=0, so ensure that there are enough available address spaces for SVCs • The node reserves VPI.VCI/0.5 for signaling (by default). • The node reserves VPI.VCI/0.16 for ILMI (by default) For the network in this example, if a 768-position range is needed for configured permanent virtual connections, and you want to reserve VPI.VCI/0.1 to VPI.VCI/0.31 either for future use or for standards, the total number of required positions in the range is 799. However, since • the number of positions must be a multiple of 256, and • the value must include the reserved position, increase the requirement to 1024 positions in the range, or decrease to 768 positions in the range. In this example, we decrease the value, thereby maximizing memory usage while maintaining a workable number of address positions. Also see remarks under “Considerations and constraints” for step 1.

(5 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
6	For the interface, determine if networking required a VCC address space under a set of contiguous non-zero VPIs.	If networking needs a range of address positions under non-zero VPIs, determine the dimensions of this range by continuing with step 6a. The number of non-zero VPIs and the range of VCIs under each VPI defines the total size of the space. To make this decision you need to have an idea of • the required range of VCC address positions for the non-zero VPI address space • how these VCC address positions are allocated to configured connections If networking does not need this range, continue with step 7. The default values for the attributes that define the non-zero VPI range are zero. In this example, networking needs the non-zero VPI range. The following paragraphs provide a general discussion on how to calculate the number of VCCs in the programmable space. **First non-zero VPI and the programmable VCC space** The number of VCCs under non-zero VPIs must be a multiple of 256. How this number is determined depends on the value of the first VPI in the range of contiguous non-zero VPIs. These rules are for calculating the number of VCCs in the connection map space. The reserved channels VCI 0-31 is not a consideration at this point.

(6 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
6	(continued)	**First non-zero VPI is 1** If the first non-zero VPI is 1, the calculation for the number of VCCs in the programmable space includes a specific number of VCCs under VPI=0. The number of VCCs in the programmable space is defined as (*nVpis* + 1) * *nVccs* where +1 in this expression represents VPI=0 The number of VCCs under VPI=0 in the calculation is equal to the number of VCCs allocated for each non-zero VPI. Determine the total number of VCCs in the non-zero space as follows: VPI=1 64 VCCs, plus VPI=2 64 VCCs, plus VPI=3 64 VCCs, plus VPI=0 64 VCCs for a total of 256 VCCs. However, only VPIs 1, 2, and 3 are available as non-zero VPI address space. The VCCs under VPI=0 in the calculation satisfy the rule that required 256 VCCs (or a multiple) under non-zero VPIs. For the purpose of calculating total VCCs in the connection map space, count the VCCs under VPI=0 once only (see step 7 for information on calculating total VCCs).

(7 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
6	(continued)	**First non-zero VPI is a multiple of 16** If the range of contiguous non-zero VPIs starts with a non-zero multiple of 16, the calculation to determine the number of VCCs in the programmable space does not include the VCCs under VPI=0. This calculation can be expressed as nVpis * nVccs For example, consider the following configuration: non-zero VPIs = 3 number of VCCs per VPI = 64 first VPI in the range = 16 The Multiservice Switch node rejects the configuration because 3 * 64 = 192, which is not a multiple of 256. A correct configuration has the number of non-zero VPIs equal to 4. Similarly, if non-zero VPIs = 4 number of VCCs per VPI = 64 first VPI in the range = 1 The Multiservice Switch node rejects the configuration because the VCCs under VPI=0 are counted in the calculation (5 * 64 = 320, which is not a multiple of 256).
		(8 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
6a	Determine how many non-zero VPIs must accommodate the VCC addresses in this space. This information is for configuring the *numNonZeroVpisForVccs* (*nVpis*) attribute under the *ConnectionMapping Override* component. **Syntax** The value defining the number of non-zero VPIs follows these rules: • The value is zero if networking does not need address positions in this range • The value is *x*, where $1 \leq x \leq 255$, if networking needs address positions in this range	If the interface needs a VCC address space under non-zero VPIs, the value of this attribute must be greater than zero. This value is part of the definition for the address range under non-zero VPIs. When a value greater than zero is configured, this address space is available for configuring connections. The size of this space is determined through the values for the *firstVpi* and *nVccs* attributes (see steps 6b and 6c). If VCC positions are not needed under non-zero VPIs, this value is zero. This address range is not required under the following conditions: • An additional range of VCC address positions is not needed on this interface. • VPI/0 is needed by third-party equipment under the interface, and therefore no VCC address position ranges can be defined for the interface. • All non-zero VPIs are needed to support third-party equipment that uses VPCs across the VPI range that is greater than 0. In configurations where a VCC address space is required, VPI values that are not assigned to carry VCCs are usable for VPCs. The calculation for the maximum number of available VPCs is 256 - # of non-zero VPIs - 1 Also see “Considerations and constraints” for step 1. For the network example in the walk-through, set this value to 3.

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
6b	Determine the range of VCC addresses that are needed under each non-zero VPI. This range is the same for all VPIs in this space. This information is for configuring the *numVccsPerNonZeroVpi* (*nVccs*) attribute under the *ConnectionMapping Override* component. **Syntax** The value defining the range of address positions under each non-zero VPI is one of 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048. If the number of non-zero VPIs defined at step 6a is zero, this value has no effect.	For this networking example, set this value to 64 (the attribute default).
6c	Determine the value of the first VPI in the range of contiguous non-zero VPIs. This information is for configuring the *firstNonZeroVpiForVccs* (*firstVpi*) attribute under the *ConnectionMapping Override* component. **Syntax** The value defining the range of address positions under each non-zero VPI is one of 1, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, 240. If the number of non-zero VPIs defined at step 6a is zero, this value has no effect.	For this networking example, set this value to 1 (the attribute default).

(10 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
7	Determine the maximum number of VCCs and VPCs that are needed under the ATM interface. This information is for configuring the *maxVccs* and *maxVpcs* attributes under the *AtmInterface* component. **Syntax** The value defining the maximum number of VCCs must be between 0 and 16 384. The value defining the maximum number of VPCs must be between 0 and 255.	The maximum number of VCCs must be less than or equal to the total number of VCC address positions reserved in the usable connection map space. This total number does not include the reserved VCI range of 0-31. In this example, the maximum number of VCCs in the connection map space is 959: 768 under VPI=0, plus 192 under the non-zero VPIs, minus 1 (for the reserved VPI.VCI/0.0). The maximum number of VCCs is expressed as maxVccs ≤ (nZVccs + (nVpis * nVccs)) - 1 Similarly, the maximum number of VPCs must be less than or equal to the total number of VPC address positions reserved in the usable connection map space. In this example, the maximum number of VPCs is 252: 256 (the total possible) less the number assigned to VCIs (four —VPIs 0, 1, 2, and 3). The maximum number of VPCs can be expressed as maxVpcs ≤ (256 - nVpis) - 1

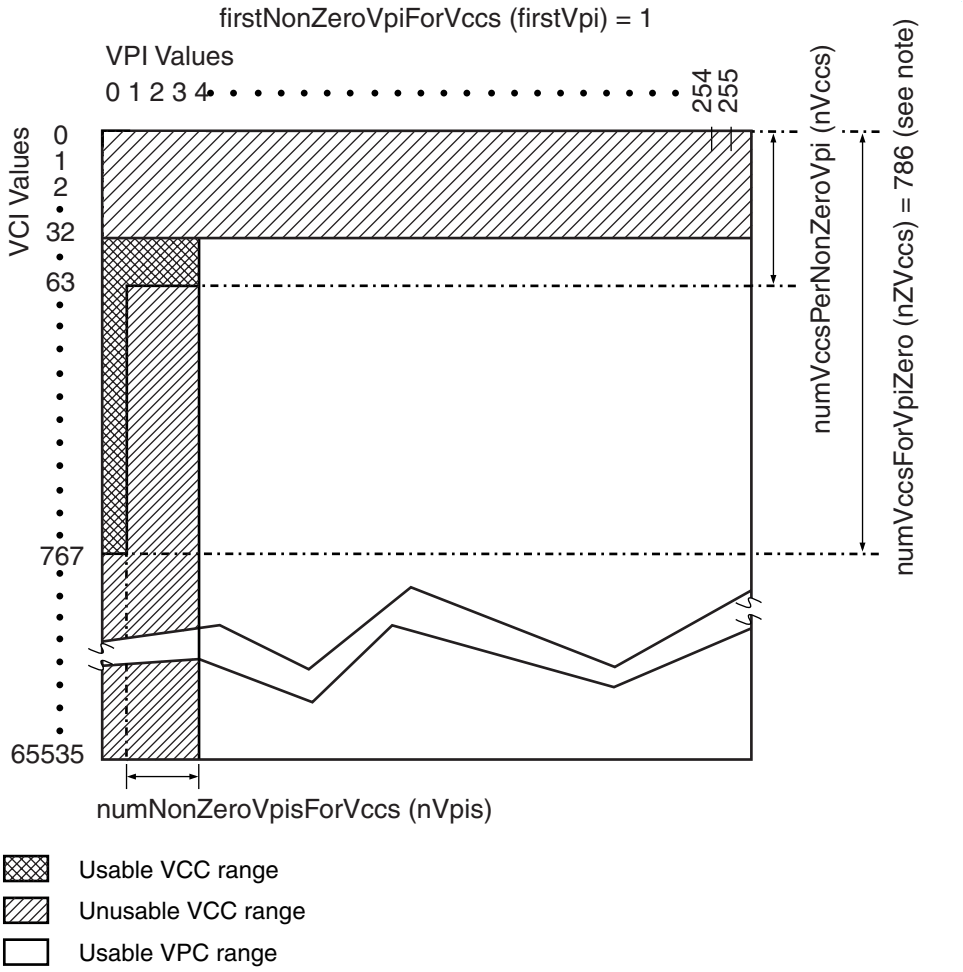
(11 of 12)

Planning example for configuring the ATM connection map for CQC-based function processors (continued)

	Step	Considerations and constraints
7	(continued)	For more information, see the following documents: • NN10600-710 *Nortel Multiservice Switch 7400/15000/20000 ATM Configuration Management* • NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals* • NN10600-706 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Shaping and Policing Fundamentals* • NN10600-707 *Nortel Multiservice Switch 7400/15000/20000 ATM Queuing and Scheduling Fundamentals* • NN10600-708 *Nortel Multiservice Switch 7400/15000/20000 ATM CAC and Bandwidth Fundamentals* Also see the *Nortel Multiservice Switch Release Notes*.
8	Translate these requirements into values that can be configured for the connection map.	The steps in this table describe how to translate the connection map requirements into configurable attributes. Here is a summary of attribute values for this example: • *connectionPoolCapacity*: 1536 (step 1) • *numVccsForVpiZero (nZVccs)*: 768 (step 5) • *numNonZeroVpisForVccs (nVpis):* 3 (step 6a) • *numVccsPerNonZeroVpi (nVccs)*: 64 (step 6b) • *firstNonZeroVpiForVccs (firstVpi)*: 1 (step 6c) • *maxVccs*: 959 (step 7) • *maxVpcs*: 252 (step 7)

(12 of 12)

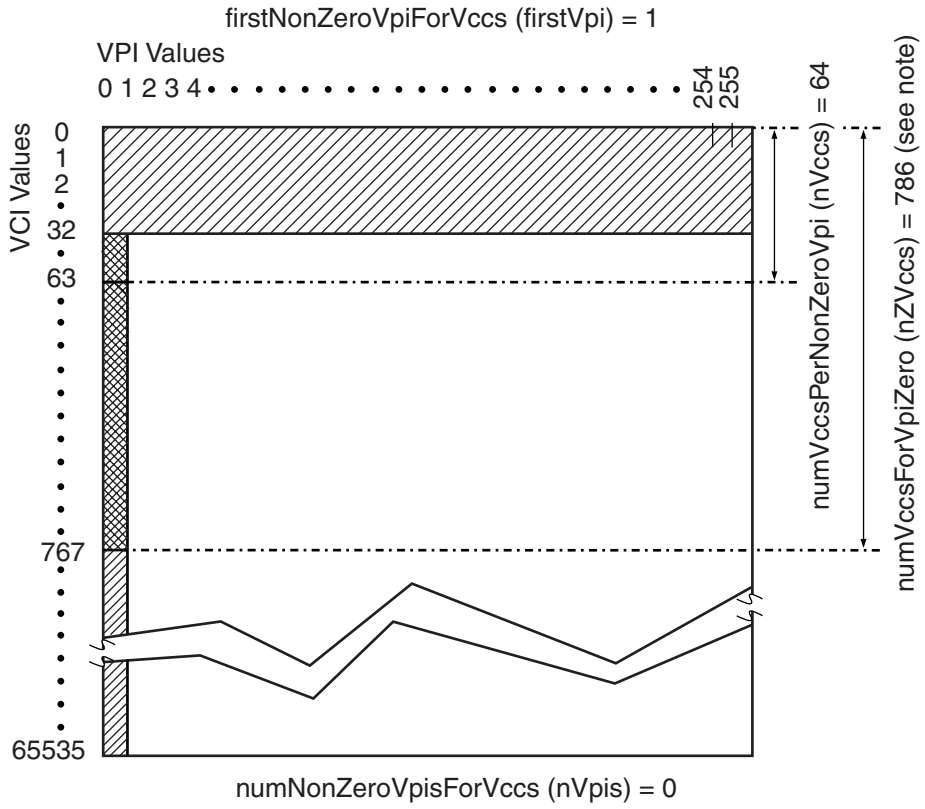
ATM interface connection map model for CQC-based function processors: walk-through example



Notes: The variable nZVccs is equal to 768 because its range is 0 to 767.
The VCI range of 0-31 is reserved for use by future standards.

PPT 2971 001 AA

ATM interface connection map for CQC-based function processors: default values



Notes: The variable nZVccs is equal to 768 because its range is 0 to 767.
The VCI range of 0-31 is reserved for use by future standards

PPT 2971 002 AA

Further planning considerations for CQC based function processors

Consider the following points when planning the network and connection map spaces on CQC-based function processors.

- Reserve some portions of the address space for industry standards that are currently under development. Although you can use addresses VPI.VCI/ 0.2 to VPI.VCI/0.31 for VCC service, it is preferable to reserve these addresses to avoid overlap with channel assignments in future standards. For example: on Nortel Multiservice Switch nodes, VPI.VCI/0.5 is the default address for SVC signaling while VPI.VCI/0.16 is the default address for integrated local management interface (ILMI). However, you can change these defaults.

- The connection map represents an address space which is a range of values. You cannot assign all addresses in the usable connection map space to VCCs. You cannot assign all remaining VPIs to VPCs.
- There are engineering limits to the number of connections that you can configure in the address space. See the *Nortel Multiservice Switch Release Notes*.

Considerations for node migration to Release 6.0

As of Release 6.0, the *ConnectionMapping* component is a dynamic optional component and is present only for 2-port and 3-port CQC-based function processors. How you migrate nodes to Release 6.0 software depends on the function processor.

For more information on function processor requirements, see the following sections:

- Considerations for 2- and 3-port CQC-based function processors (page 225)
- Considerations for 8-port CQC-based function processors (page 225)

Considerations for 2- and 3-port CQC-based function processors

When you migrate a node to Release 6.0 or later, the upgrade process configures the *ConnectionMapping* component depending on the application of existing Nortel Multiservice Switch default values for the prior release.

If the existing *ConnectionMapping* component configuration is based on default values, the process does not create a configured instance of the *ConnectionMapping* component. The node uses dynamic component values for its attributes. If the existing *ConnectionMapping* component configuration is not based on Multiservice Switch default values, the upgrade process creates the *Override* subcomponent and all attributes from the existing *ConnectionMapping* component migrate to the *Override* subcomponent.

In either scenario, the upgrade process creates the operational instance of the *ConnectionMapping* component to permit operator access to the operational attributes.

Considerations for 8-port CQC-based function processors

When you migrate a node to Release 6.0 or later, the upgrade process deletes the *ConnectionMapping* component for any 8-port CQC-based function processors on the node. The values of the attributes do not migrate to the new release, because the Nortel Multiservice Switch node does not use *ConnectionMapping* component, either configured or dynamic, for the function processor types.

You do not need to reconfigure for the removed *ConnectionMapping* component or intervene in the upgrade process. However, can set the number of active VPI bits through the *maxVpiBits* attribute under the *AtmIf* component. See Connection map address assignment for CQC function processors (page 58).

Compliance statements

Use the following sections to learn more about Nortel Multiservice Switch compliance to standards for ATM networking, including routing and signaling.

Navigation

Compliance terminology

The following terminology appears in the tables:

- Noted. This terminology applies where the specification provides clarification, non-specific information, or details that do not relate directly to the ATM interface.
- Fully compliant. This terminology applies where Multiservice Switch ATM functionality fully complies with the text for this section.
- Compliant with these exceptions. This terminology applies where the Multiservice Switch ATM interface does not completely comply with the text. The text indicates exceptions.
- Not supported. This terminology applies where the Multiservice Switch ATM interface does not support the functionality described in the text.

Compliance with ATM Forum UNI 3.x specifications

Nortel Multiservice Switch ATM implementation is compliant with the following standards, with some exceptions:

- *User-to-Network Interface Specification Version 3.0* (af-uni-0010.001)

- *User-to-Network Interface Specification Version 3.1* (af-uni-0010.002)

The table Compliance with UNI 3.x specification (page 228) identifies the exceptions.

Compliance with UNI 3.x specification

Section	Section title	Compliance
Section 1.1	Purpose of Document	Noted
Section 1.2	Scope of Document	Noted
Section 1.3	Structure of Document	Noted
Section 1.4	Terminology	Noted
Section 1.5	ATM Bearer Service Overview	Fully complies with
Section 1.6	User-Network Interface Configuration	Noted
Section 1.7	User-Network Interface Protocol Architecture	Fully complies with
Section 2	ATM Physical Layer Interface Specification	Fully complies with
Section 3.1	ATM Layer Service	Fully complies with
Section 3.2	Service Expected from the Physical Layer	Fully complies with
Section 3.3	ATM Cell Structure and Encoding at the UNI	Fully complies with
Section 3.4	ATM Layer Functions Involved at the UNI (U-plane)	Fully complies with
Section 3.5	ATM Layer Management Specification (M-plane)	Fully complies with
Section 3.6	Traffic Control and Congestion Control	Fully complies with
Section 4	Interim Local Management Interface Specifications	Compliant with these exceptions: Requests for management information other than dynamic address registration as defined in the “ATM-FORUM-ADDR-REG” MIB cause a noSuchName response to be sent back to the originator.
Section 5	UNI signaling	Fully complies with

Compliance with ATM Forum UNI 4.0 specifications

Nortel Multiservice Switch ATM implementation is compliant with all mandatory sections of the *User-to-Network Interface Specification Version 4.0* (af-sig-0061.000).

Compliance with ATM Forum ILMI 4.0 specifications

Nortel Multiservice Switch ATM implementation is compliant with all mandatory sections of the *Integrated Local Management Interface (ILMI) Specification Version 4.0* (af-ilmi-0065.000).

Compliance with ATM Forum AINI 1.0 specifications

Nortel Multiservice Switch ATM implementation is compliant with all mandatory sections of the *ATM Inter-Network Interface (AINI) Specification Version 1.0* (af-cs-0125.000).

Compliance with ATM Forum PNNI 1.0 specifications

Nortel Multiservice Switch ATM implementation is compliant with the following standards, with some exceptions:

- *Private Network-Network Interface (PNNI) Specification Version 1.0* (af-pnni-0055.000)
- *PNNI V1.0 Errata and PICS* (af-pnni-0081.000)

The tables PNNI 1.0 mandatory features supported on Multiservice Switch (page 229) and PNNI 1.0 optional features supported on Multiservice Switch (page 232) describe the ATM networking system capabilities based on the mandatory and optional capabilities list in Annex G of the ATM Forum *Private Network-Network Interface (PNNI) Specification Version 1.0* with Annex G changes incorporated as specified in the *PNNI V1.0 Errata and PICS*.

Multiservice Switch ATM networking includes the following value-added features:

- accounting on PNNI Interfaces (SVCs and SPVCs)
- interworking of UNI 3.1, IISP 1.0 based on UNI 3.1 and PNNI 1.0 signaling
- SPVC interworking between UNI 3.1, IISP 1.0 based on UNI 3.1 and PNNI 1.0
- ATM Forum Traffic Management 4.0 subset
- X.121 based NSAP addressing support on PNNI and IISP interfaces

PNNI 1.0 mandatory features supported on Multiservice Switch

Capability	Description	Supported in this release
Support inside links	Support for links within peer groups	Yes
Version negotiation	Determine a common version if neighboring PNNI nodes are running different sets of PNNI versions	Yes

PNNI 1.0 mandatory features supported on Multiservice Switch (continued)

Capability	Description	Supported in this release
Information group tags	PNNI packets are built using information groups, identified by their corresponding group tags	Yes
Hello protocol over inside links	Staging of links within peer groups	Yes
Database synchronization	Exchange of summary of topology information in store	Yes
Flooding	Exchange of actual topology information (in the form of PNNI topology state elements) by a reliable mechanism	Yes
Understand all defined PTSE types	Ability to process nodal state, nodal information, exterior reachable, interior reachable, horizontal, and uplink PNNI topology state elements (PTSEs)	Yes
Origination of nodal information PTSEs	Capability to originate PTSEs describing non-state information about the node (for example, node ID)	Yes
Origination of horizontal link PTSEs	Capability to originate PTSEs describing state of a horizontal link (for example, available cell rate)	Yes
Origination of internal reachable address PTSEs	Capability to originate PTSEs describing a reachable address, with or without details of resources available to get to it	Yes
Vote in PGL elections	Vote in election of peer group leader	Yes
Perform default internal address summarization	Capability to identify addresses for which a summary address is defined and then advertising only the summary address or suppressing every such match	Yes
Point-to-point calls	Support for connections between two ATM end systems. These connections are bidirectional or unidirectional.	Yes
Point-to-multipoint calls	Support for connections between a single source end-station (root node) and multiple destination end-stations (known as leaves). These connections are unidirectional.	Yes
Signaling of individual QoS parameters	Support for extended QoS IE and End-to-End Transit Delay IE	Yes

(2 of 3)

PNNI 1.0 mandatory features supported on Multiservice Switch (continued)

Capability	Description	Supported in this release
ATM anycast	Use of addresses to identify services rather than a particular node (for example, a call to an anycast address is routed to the nearest end-system that registered itself with the network to provide the associated service)	Yes
Generate a multi-level DTL	Generate DTLs that traverse more than one peer group	Yes
Follow a multi-level DTL	Use a DTL that traverses more than one peer group	Yes
Crankback	Capability to retract call setup back to the originator or last entry border node if the call setup cannot be routed any further	Yes
Outside link support	Links between different peer groups	Yes
Hello protocol over outside links	Able to run the Hello protocol between nodes in a different peer group	Yes
Originate uplink PTSEs	PTSE describing uplinks to higher level nodes	Yes
Entry and exit border node DTL handling	Append and remove DTL IE instances at entry and exit border nodes	Yes
Entry border node crankback handling	Attempt alternate routes and modify crankback IE if necessary	Yes
SVC-based RCC	Create SVC connections between LGNs for use as a RCC channel	Yes
SVC-based RCC Hello protocol	Hello protocol used to monitor the RCC channel	Yes
LGN Horizontal Link Hello protocol	Hello protocol used to monitor logical links between LGNs	Yes
PGL capable	Be the peer group leader of a peer group.	Yes
Link aggregation	Combine lower level links at a higher level to reduce complexity	Yes
Nodal aggregation	Combine multiple nodes in a peer group to be represented as a single node	Yes
Interior and exterior address summarization	Summary of summary addresses	Yes

(3 of 3)

PNNI 1.0 optional features supported on Multiservice Switch

Capability	Description	Supported in this release
Origination of exterior reachable addresses advertisement	Ability to originate PTSEs describing exterior reachable addresses	Yes
Alternate routing as a result of crankback	Performing alternate routing when a call blocks at a node or link along the path and cranks back	Yes
Hello protocol over VPCs	Useful for peer groups split across a public network	No
Associated signaling	VP associated signaling	Partial support (see [1])
Negotiation of ATM traffic descriptors	Support for traffic parameter negotiation	No
Switched VP service	Ability to support switched VPs over PNNI links	Yes (see [2])
Soft PVPC and PVCC support	Support for Soft PVPC and PVCC establishment over PNNI links.	Yes
ABR signalling for point-to-point calls	Support for ABR service category	No
Generic ID transport	Support the Generic Identifier Transport IE over PNNI links	Yes
Frame discard	Support for frame discard traffic management option	Yes
ILMI over PNNI links	Support ILMI channel over PNNI links	No
[1]Multiservice Switch allows ‘VP based’ signaling using the virtual UNI capability		
[2] Multiservice Switch loads preceding PCR4.2 do not support VP capability over PNNI links. Due to backward compatibility issues with these loads, Multiservice Switch may route to switches that do not support VP associated signaling causing the call to crankback, or in cases where the maximum number of crankbacks has been reached, fail.		

Compliance with ATM Forum TM 4.0 specifications

In general, Nortel Multiservice Switch FPs are compliant to TM 4.0 except for ABR. See NN10600-705 *Nortel Multiservice Switch 7400/15000/20000 ATM Traffic Management Fundamentals* for more information.

Compliance with draft ITU-T standards

Nortel Multiservice Switch ATM implementation is compliant with the following standards and recommendations:

- Q.2100
- Q.2110 (Multiservice Switch compliance is based on ATM UNI 3.1)

- Q.2130 (Multiservice Switch compliance is based on ATM UNI 3.1)
- Q.2610 (Multiservice Switch compliance is based on ATM UNI 3.0 and 3.1)
- Q.2931 (Multiservice Switch compliance is based on ATM UNI 3.0 and 3.1)
- Q.2961.1 (Multiservice Switch compliance is based on ATM UNI 3.0 and 3.1)
- Q.2961.2 (Multiservice Switch compliance is based on ATM UNI 3.1 - only QoS class 0 is defined for compatibility with ITU-T)

Multiservice Switch is compliant with ITU-T draft standards on signaling (including Q.2931, Q.2110, and Q2130) to the same extent that the ATM Forum UNI Versions 3.0 and 3.1 are compliant. The list of the ATM Forum UNI 3.1 signaling differences is attached in Appendix E of the *ATM User-to-Network Interface Specification Version 3.1* (ATM Forum Technical Committee).

Implementation of SSCF and SSCOP used on Multiservice Switch for ATM Forum UNI Version 3.1 signaling is fully compliant with ITU-T draft versions of ITU-T Q.2110 and Q.2130.

Attention: ATM UNI Version 3.0 is based on early draft version of the Q.SAAL.1 and Q.SAAL.2 specification, as well as Q.93b. It is not compliant with ITU-T Q.2110, Q.2130, and Q.2610.

Multiservice Switch handling of OAM cells (AIS, RDI, loopbacks) complies with ITU-T I.610.

Multiservice Switch fault management system does not support performance monitoring or continuity check cell types.

Nortel Multiservice Switch 7400/15000/20000

ATM Routing and Signalling Fundamentals

Publication: NN10600-702
Document status: Standard
Document issue: 7.2S1
Document date: March 2006
Product release: PCR7.2 and up
Job function: Product Fundamentals
Type: NTP
Language type: U.S. English