ATM Switch Network Configuration Manual

MANU0148-06
Revision A
March 5, 1999

FORE
SYSTEMS®

Software Version 6.0.x

FORE Systems, Inc.

1000 FORE Drive
Warrendale, PA 15086-7502
Phone: 724-742-4444
FAX: 724-742-7742

http://www.fore.com

Legal Notices

U.S. Government Restricted Rights. If you are licensing the Software on behalf of the U.S. Government ("Government"), the following provisions apply to you. If the Software is supplied to the Department of Defense ("DoD"), it is classified as "Commercial Computer Software" under paragraph 252.227-7014 of the DoD Supplement to the Federal Acquisition Regulations ("DFARS") (or any successor regulations) and the Government is acquiring only the license rights granted herein (the license rights customarily provided to non-Government users). If the Software is supplied to any unit or agency of the Government other than DoD, it is classified as "Restricted Computer Software" and the Government's rights in the Software are defined in paragraph 52.227-19 of the Federal Acquisition Regulations ("FAR") (or any successor regulations) or, in the cases of NASA, in paragraph 18.52.227-86 of the NASA Supplement to the FAR (or any successor regulations).

Printed in the USA.

This publication is provided by FORE Systems, Inc. "as-is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties or conditions of merchantability or fitness for a particular purpose. FORE Systems, Inc. shall not be liable for any errors or omissions which may occur in this publication, nor for incidental or consequential damages of any kind resulting from the furnishing, performance, or use of this publication.

Information published here is current or planned as of the date of publication of this document. Because we are improving and adding features to our products continuously, the information in this document is subject to change without notice.

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 (October 1988) and FAR 52.227-19 (June 1987).

The VxWorks software used in the Mini Loader is licensed from Wind River Systems, Inc., Copyright ©1984-1998.

TRADEMARKS

FORE Systems, AVA, *ForeRunner*, *ForeThought*, *ForeView*, and *PowerHub* are registered trademarks of FORE Systems, Inc. All Roads Lead To ATM, ASN, ATV, *CellChain*, *CellPath*, *CellStarter*, *EdgeRunner*, *FramePlus*, *ForeRunnerHE*, *ForeRunnerLE*, Intelligent Infrastructure, I2, MSC, NetPro, Networks Of Steel, *StreamRunner*, TNX, Universal Port, *VoicePlus*, and Zero Hop Routing are unregistered trademarks of FORE Systems, Inc. All other brands or product names are trademarks or registered trademarks of their respective holders.

FCC CLASS A NOTICE

WARNING: Changes or modifications to this unit not expressly approved by the party responsible for compliance could void this user's authority to operate this equipment.

NOTE: The *TNX*-210, *TNX*-1100, *ASX*-200WG, *ASX*-200BX, *ASX*-1000, *ASX*-1200, *ASX*-4000, *ForeRunnerLE* 25, and *ForeRunnerLE* 155 have been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of the equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

DOC CLASS A NOTICE

This digital apparatus does not exceed Class A limits for radio noise emission for a digital device as set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le present appareil numerique n’emet pas de bruits radioelectriques depassant les limites applicables aux appareils numeriques de la class A prescrites dans le reglement sur le brouillage radioelectrique edicte par le ministere des Communications du Canada.

CE NOTICE

Marking by the symbol **CE** indicates compliance of this system to the EMC (Electromagnetic Compatibility) directive of the European Community and compliance to the Low Voltage (Safety) Directive. Such marking is indicative that this system meets or exceeds the following technical standards:

- EN 55022 - “Limits and Methods of Measurement of Radio Interference Characteristics of Information Technology Equipment.”
- EN 50082-1 - “Electromagnetic compatibility - Generic immunity standard Part 1: Residential, commercial, and light industry.”
- IEC 1000-4-2 - “Electromagnetic compatibility for industrial-process measurement and control equipment Part 2: Electrostatic discharge requirements.”
- IEC 1000-4-3 - “Electromagnetic compatibility for industrial-process measurement and control equipment Part 3: Radiate electromagnetic field requirements.”
- IEC 1000-4-4 - “Electromagnetic compatibility for industrial-process measurement and control equipment Part 4: Electrical fast transient/burst requirements.”

VCCI CLASS A NOTICE

　この装置は、情報処理装置等電波障害自主規制協議会（ＶＣＣＩ）の基準に基づくクラスＡ情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

This is a Class A product based on the standard of the Voluntary Control Council for Interference by Information Technology Equipment (VCCI). If this equipment is used in a domestic environment, radio disturbance may arise. When such trouble occurs, the user may be required to take corrective actions.

AUSTRALIA EMC COMPLIANCE

This product has been tested and found to comply with the Class A electromagnetic compatibility limits specified in AS/NZ 3548.

FCC REQUIREMENTS (Notice to Users of DS1 Service)

The following instructions are provided to ensure compliance with the FCC Rules, Part 68.

(1) This device must only be connected to the DS1 network connected behind an FCC Part 68 registered channel service unit. Direct connection is not allowed.

(2) Before connecting your unit, you must inform the telephone company of the following information:

Port ID	REN/SOC	FIC	USOC
NM-6/DS1C NM-2/DS1C NM-8/DS1D NM-4/DS1D	6.0N	04DU9-BN, 04DU9-DN, 04DU9-1ZN 04DU9-1SN, and 04DU9-1KN	RJ48C

(3) If the unit appears to be malfunctioning, it should be disconnected from the telephone lines until you learn if your equipment or the telephone line is the source of the trouble. If your equipment needs repair, it should not be reconnected until it is repaired.

(4) If the telephone company finds that this equipment is exceeding tolerable parameters, the telephone company can temporarily disconnect service, although they will attempt to give you advance notice if possible.

(5) Under the FCC Rules, no customer is authorized to repair this equipment. This restriction applies regardless of whether the equipment is in or out of warranty.

(6) If the telephone company alters their equipment in a manner that will affect use of this device, they must give you advance warning so as to give you the opportunity for uninterrupted service. You will be advised of your right to file a complaint with the FCC.

CANADIAN IC CS-03 COMPLIANCE STATEMENT

NOTICE: The Industry Canada label identifies certified equipment. This certification means that the equipment meets certain telecommunications network protective, operational and safety requirements. The Industry Canada label does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. In some cases, the company's inside wiring associated with a single line individual service may be extended by means of a certified connector assembly (telephone extension cord). The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be made by an authorized Canadian maintenance facility designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.

Caution: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.

E1 AND E3 NOTICE

The E1 (NM-6/E1C and NM-2/E1C) and E3 (NM-4/E3C and NM-2/E3C) network modules that are described in this manual are approved for use in FORE Systems' host systems providing that the instructions below are strictly observed. Failure to follow these instructions invalidates the approval.

Pan European Approval - CE Marking

Pan European approval of the E1 network module was issued by BABT following assessment against CTR12. This means that it can be connected to ONP and unstructured PTO-provided private circuits with 120 Ω interfaces in all European countries, according to Telecommunications Terminal Equipment (TTE) Directive 91/263/EEC. Thus, the following CE mark applies:

CE168X

The E1 and E3 network modules conform to safety standard EN60950 1992 following the provisions of Low Voltage Product Safety Directive 73/23/EEC and CE Marking Directive 93/68/EEC, and can be marked accordingly with the CE symbol.

The E1 and E3 network modules conform to EN55022 1994 and EN50082-1 1992 following the provisions of the EMC Directive 89/336/EEC, and can be marked accordingly with the CE symbol.

National Approvals

UK

Network Module	Connects to	Approval Number
E1	Structured and unstructured PTO-provided private circuits with 75 Ω interfaces	AA60953
E3	PTO-provided private circuits with 75 Ω interfaces	NS/4387/1/T/605954

Germany

Network Module	Connects to	Approval Number
E3	Structured PTO-provided private circuits with 75 Ω interfaces	A127535H for the ASX-1000 A127534H for the ASX-200BX or ASX-200WG

Switzerland

Network Module	Connects to	Approval Number
E1	Structured PTO-provided private circuits with 120 Ω interfaces	96.0872.J.N
E3	Structured PTO-provided private circuits with 75 Ω interfaces	96.0873.J.N

Required User Guide Statements - UK Installation

The use of auxiliary products not authorized by FORE Systems® in FORE Systems ATM Switches may cause the power specification to be exceeded and is a potential safety hazard.

The equipment must be installed such that with the exception of the connections to the host, clearance and creepage distances shown in the table below are maintained between the network module and any other assemblies which use or generate a voltage shown in the table below. The larger distance shown in brackets applies where the local environment within the host is subject to conductive pollution or dry non-conductive pollution which could become conductive due to condensation. Failure to maintain these minimum distances invalidates the approval.

Clearance (mm)	Creepage (mm)	Voltage Used or Generated by Host or by Network Modules
2.0	2.4 (3.8)	Up to 50 V_{rms} or V_{dc}
2.6	3.0 (4.8)	Up to 125 V_{rms} or V_{dc}
4.0	5.0 (8.0)	Up to 250 V_{rms} or V_{dc}
4.6	6.4 (10.0)	Up to 300 V_{rms} or V_{dc}

For a host or other expansion card fitted in the host, using or generating voltages greater than 300V (rms or dc), advice from a competent telecommunications engineer must be obtained before installation of the relevant equipment.	Above 300 V_{rms} or V_{dc}

NOTE: Installing the network modules in the appropriate FORE Systems hosts, according to the installation instructions provided, satisfies the requirements listed above.

The following tables show the available ports and their safety status:

NM-6/E1C and NM-2/E1C

Ports	Safety Status
E1 Ports	TNV operating at SELV
Bus Connector	SELV

NM-4/E3C and NM-2/E3C

Ports	Safety Status
E3 Ports	TNV operating at SELV
Bus Connector	SELV

SAFETY CERTIFICATIONS

ETL certified to meet Information Technology Equipment safety standards UL 1950, CSA 22.2 No. 950, and EN 60950.

Table of Contents

CHAPTER 2 Configuring Classical IP

CHAPTER 3 Configuring an Emulated LAN

CHAPTER 4 MPOA

CHAPTER 5 *ForeThought* PNNI

CHAPTER 7 Signalling

CHAPTER 8 Security

APPENDIX C Converting FT-PNNI and PNNI Networks

APPENDIX D Configuring *FramePlus* Modules

Acronyms

Glossary

Index

List of Figures

CHAPTER 1 **Configuring PVCs**

CHAPTER 2 **Configuring Classical IP**

CHAPTER 3 **Configuring an Emulated LAN**

CHAPTER 4 MPOA

CHAPTER 5 *ForeThought* PNNI

CHAPTER 6 ATM Forum PNNI

CHAPTER 7 Signalling

APPENDIX C Converting FT-PNNI and PNNI Networks

APPENDIX D Configuring *FramePlus* Modules

List of Tables

Preface

This manual provides the technical information needed to configure the *ForeRunner*® family of ATM Switches, *ForeRunnerLE*® Switches, *TNX*™ ATM Switches, ESX™-3000 Switch, the *ForeRunner* network modules, and the accompanying *ForeThought*™ software. This document also provides general ATM information and general product information. This document was created for users with various levels of experience. If you have any questions or problems, please contact the FORE Systems' Technical Assistance Center (TAC).

Chapter Summaries

Chapter 1 - Configuring PVCs - Describes how to create PVCs on a switch through the ATM Management Interface (AMI).

Chapter 2 - Configuring Classical IP - Describes how to design, configure, and maintain a Classical IP ATM network.

Chapter 3 - Configuring an Emulated LAN - Provides an overview of LAN Emulation and gives an example of how to configure an Emulated LAN.

Chapter 4 - MPOA - Contains an overview of LAN Emulation (LANE) and Multi-Protocol Over ATM (MPOA).

Chapter 5 - ForeThought PNNI - Provides an overview of *ForeThought* PNNI and shows how this scalable routing and signalling protocol can be used to simplify large network topologies.

Chapter 6 - ATM Forum PNNI - Provides an overview of ATM Forum PNNI and shows how this scalable routing and signalling protocol can be used to simplify large network topologies.

Chapter 7 - Signalling - Describes signalling protocol information.

Chapter 8 - Security - Describes the various forms of security that can be used on the switch.

Chapter 9 - Configuring Timing - Describes how to set up timing on a switch.

Appendix A - Configuring SNMP - Describes the remote SNMP configuration of a switch.

Appendix B - Configuring Circuit Emulation Services - Contains information for configuring Circuit Emulation Services (CES) network modules.

Appendix C - Converting FT-PNNI and PNNI Networks - Discusses the conversion of both non-hierarchical and hierarchical FT-PNNI networks to ATM Forum PNNI networks. Also describes the conversion of a non-hierarchical PNNI network to a lop-sided hierarchical PNNI network.

Appendix D - Configuring FramePlus Modules - Contains information for configuring *FramePlus*™ network modules.

Acronyms - Contains a list of common networking acronyms.

Glossary - Contains definitions for networking terms, both general and ATM-specific.

Related Manuals

References are made in this manual to the following manuals:

AMI Configuration Command Reference Manual, Part 1 - Describes the configuration level AMI commands and menus from **`configuration alarms>`** to **`configuration nsap>`**.

AMI Configuration Command Reference Manual, Part 2 - Describes the configuration level AMI commands and menus from **`configuration port>`** to **`configuration vpt>`**.

ATM Management Interface (AMI) Manual - Describes the root, display, operation, and statistics level AMI commands and menus.

ATM Switch Diagnostics and Troubleshooting Manual - Describes the debug level AMI commands and menus. Also, describes error messages, loopbacks, SCP diagnostics, and ATM Forum PNNI debugging information.

These manuals can be found on the CD and can be read and printed using Acrobat Reader which is also included on the CD. If Acrobat Reader is installed locally, run Acrobat and open the manual from the `/DOCS` directory of the CD. If Acrobat Reader is not installed locally, run the Acrobat installer to load Acrobat Reader on your machine. Then run the `ACROREAD.EXE` file in the `/DOCS` directory of the CD.

Technical Support

In the U.S.A., customers can reach FORE Systems' Technical Assistance Center (TAC) using any one of the following methods:

1. Select the "Support" link from FORE's World Wide Web page:

   http://www.fore.com/

2. Send questions, via e-mail, to:

   support@fore.com

3. Telephone questions to "support" at:

   800-671-FORE (3673) or **724-742-6999**

4. FAX questions to "support" at:

   724-742-7900

Technical support for customers outside the United States should be handled through the local distributor or via telephone at the following number:

+1 724-742-6999

No matter which method is used to reach FORE Support, customers should be ready to provide the following:

- A support contract ID number
- The serial number of each product in question
- All relevant information describing the problem or question

Typographical Styles

Throughout this manual, all specific commands meant to be entered by the user appear on a separate line in bold typeface. In addition, use of the `Enter` or `Return` key is represented as **`<ENTER>`**. The following example demonstrates this convention:

`cd /usr <ENTER>`

File names that appear within the text of this manual are represented in the following style: "...the `fore_install` program installs this distribution."

Command names that appear within the text of this manual are represented in the following style: "...using the **`flush-cache`** command clears the bridge cache."

Subsystem names that appear within the text of this manual are represented in the following style: "...to access the **`bridge`** subsystem..."

Parameter names that appear within the text of this manual are represented in the following style: "...using *`<seg-list>`* allows you to specify the segments for which you want to display the specified bridge statistics."

Any messages that appear on the screen during software installation and network interface administration are shown in `Courier` font to distinguish them from the rest of the text as follows:

```
.... Are all four conditions true?
```

Important Information Indicators

To call your attention to safety and otherwise important information that must be reviewed to ensure correct and complete installation, as well as to avoid damage to the FORE Systems product or to your system, FORE Systems utilizes the following *WARNING/CAUTION/NOTE* indicators.

WARNING statements contain information that is critical to the safety of the operator and/or the system. Do not proceed beyond a ***WARNING*** statement until the indicated conditions are fully understood or met. This information could prevent serious injury to the operator, damage to the FORE Systems product, the system, or currently loaded software, and is indicated as follows:

WARNING!

Hazardous voltages are present. To reduce the risk of electrical shock and danger to personal health, follow the instructions carefully.

CAUTION statements contain information that is important for proper installation/operation. Compliance with **CAUTION** statements can prevent possible equipment damage and/or loss of data and are indicated as follows:

CAUTION

You risk damaging your equipment and/or software if you do not follow these instructions.

NOTE statements contain information that has been found important enough to be called to the special attention of the operator and is set off from the text as follows:

NOTE

If you change the value of the LECS control parameters while the LECS process is running, the new values do not take effect until the LECS process is stopped, and then restarted.

Laser Warning

Every FORE Systems network module having a single mode fiber optic interface contains a Class 1 laser.

Class 1 Laser Product:
This product conforms to applicable requirements of 21 CFR 1040 at the date of manufacture.

Class 1 lasers are defined as products which do not permit human access to laser radiation in excess of the accessible limits of Class 1 for applicable wavelengths and durations. These lasers are safe under reasonably foreseeable conditions of operation.

WARNING!

Do not stare into the beam or view the beam with optical instruments.

Safety Precautions

For your protection, observe the following safety precautions when setting up equipment:

- Follow all warnings and instructions marked on the equipment.
- Ensure that the voltage and frequency of your power source matches the voltage and frequency inscribed on the equipment’s electrical rating label.
- Never push objects of any kind through openings in the equipment. Dangerous voltages may be present. Conductive foreign objects could produce a short circuit that could cause fire, electric shock, or damage to your equipment.

Modifications to Equipment

Do not make mechanical or electrical modifications to the equipment. FORE Systems, Inc., is not responsible for regulatory compliance of a modified FORE product.

Placement of a FORE Systems Product

CAUTION

To ensure reliable operation of your FORE Systems product and to protect it from overheating, openings in the equipment must not be blocked or covered. A FORE Systems product should never be placed near a radiator or heat register.

Power Cord Connection

WARNING!

FORE Systems products are designed to work with single-phase power systems having a grounded neutral conductor. To reduce the risk of electrical shock, do not plug FORE Systems products into any other type of power system. Contact your facilities manager or a qualified electrician if you are not sure what type of power is supplied to your building.

WARNING!

Your FORE Systems product is shipped with a grounding type (3-wire) power cord. To reduce the risk of electric shock, always plug the cord into a grounded power outlet.

CHAPTER 1 Configuring PVCs

To establish a permanent communication link between two sites, it is necessary to establish permanent virtual circuits (PVCs) at each switch along the communications path. This chapter discusses the creation of PVCs through the ATM Management Interface (AMI), a command-line user interface to *ForeRunner*, *ForeRunnerLE*, *TNX*, and *ESX*-3000 ATM switches.

1.1 General Concepts

Each ATM cell contains a virtual path identifier (VPI) and a virtual channel identifier (VCI) as part of its five-byte ATM header. The VPI and VCI are used to route the cell through the ATM network. When a switch fabric receives a cell, it examines the ATM header to determine the correct output port, VPI, and VCI for the cell. For example, an ATM switch fabric can be configured such that any cell received on port A1 with VPI|VCI = 0|32 is switched to port B2 with VPI|VCI = 0|35. The translation from input port, VPI, and VCI to output port, VPI, and VCI is achieved via a mapping table in the switch fabric's memory.

The VCI value of cells does not change as the cell is switched through the ATM network via a virtual path. In a single switch environment, a cell's VPI and VCI are translated only once, but in a multiple switch environment a cell's VPI and VCI are translated many times. It is important to remember that a cell's VPI and VCI are of local significance only (i.e., link-by-link). It is also important to note that virtual connections are unidirectional; that is, they are valid in one direction only. The VPI and VCI **may** change as the cell is switched through the network.

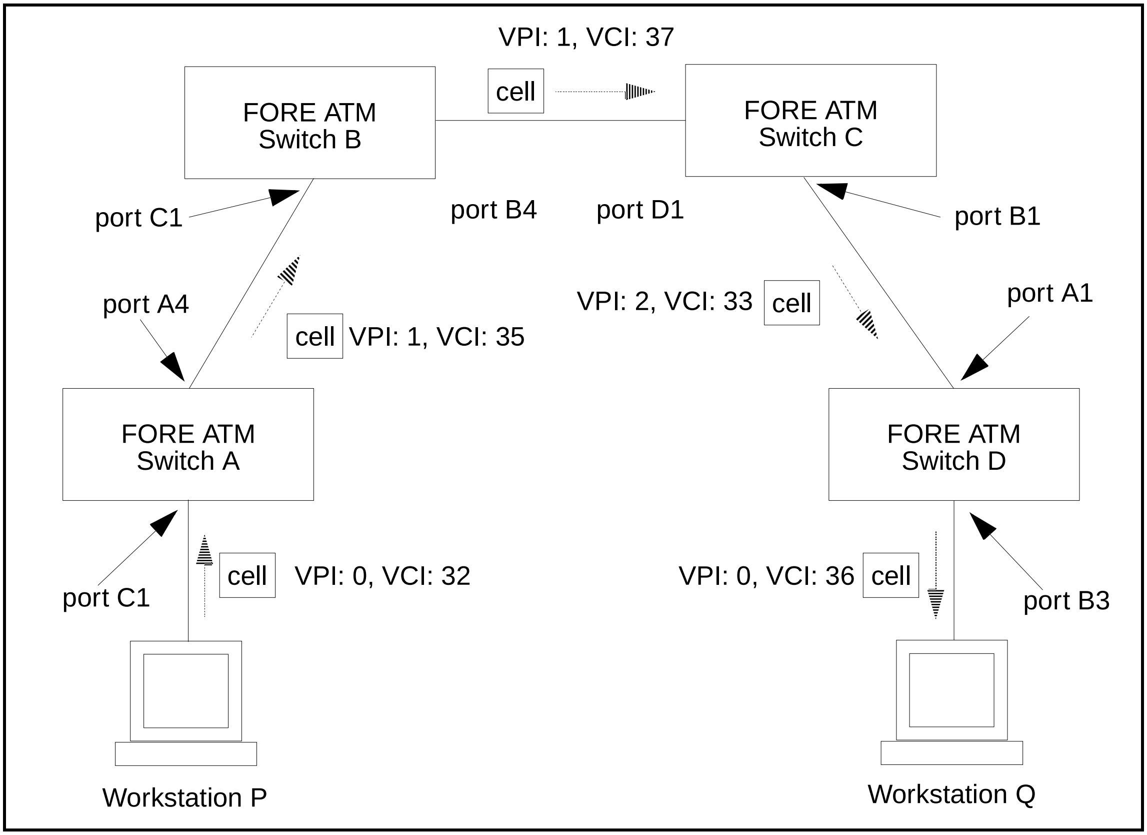

Figure 1.1 - The Cell

The mappings in an ATM network used to route cells from a source to a destination are generally referred to as virtual channels and virtual paths. The following sections is to explain how to create the necessary mappings to establish these virtual paths and virtual channels in a network of FORE ATM switches.

1.2 Virtual Paths

Virtual paths, which are carried within a physical transit medium (e.g., DS1, E1, DS3, E3, OC3c, or OC12c link), are used to establish connections between two nodes in an ATM network. Many virtual paths can be transmitted within a single physical link. Two types of virtual paths exist: virtual path connections (VPCs), also known as through paths, and originating/terminating paths, also known as virtual path terminators (VPTs). VPCs allow virtual paths to be cross-connected at a switch node while VPTs allow virtual channels (VCCs) to be cross-connected or switched at a switch node.

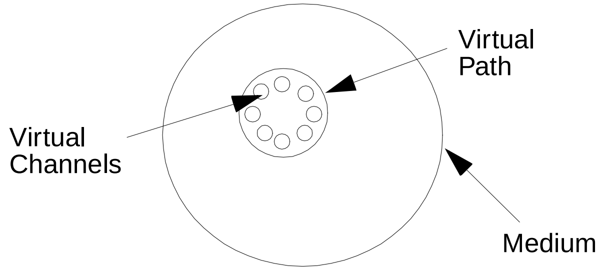

Figure 1.2 - Virtual Channels in a Virtual Path

A single virtual path can be used to route many virtual channels through the ATM network. Because a virtual path simply routes virtual channels through the network, a cell is guaranteed to have the same VCI when it exits the virtual path as it had when it entered the virtual path.

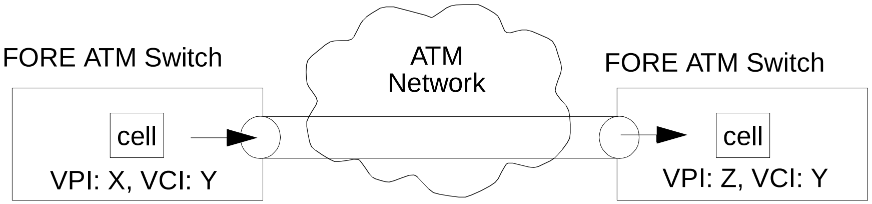

Figure 1.3 - An Example of a Virtual Path

The VCI value of cells does not change as the cell is switched through the ATM network via a virtual path. Each virtual path must originate at a switch fabric, pass through zero or more switch fabrics and terminate at another switch fabric. The origination and termination points are referred to as originating and terminating paths. Virtual paths are switched through switch fabrics via through paths. Virtual paths are made up of an originating path, zero or more through paths, and a terminating path.

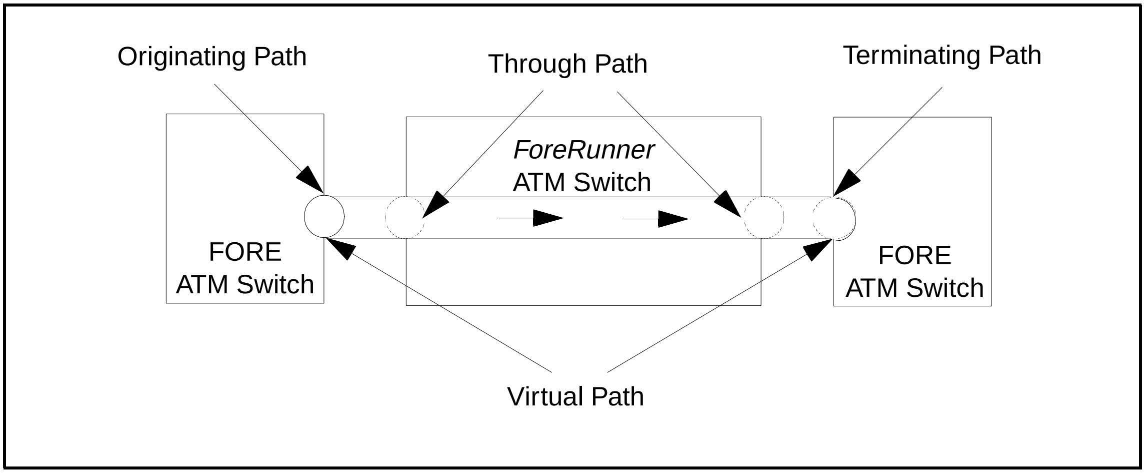

Figure 1.4 - Composition of a Virtual Path

1.2.1 Through Paths

Through paths route an entire virtual path through an ATM switch fabric. When a cell is received by a switch fabric on a through path, the VPI is examined to determine the output port and VPI. The VCI component of the ATM header remains unchanged and can have any value. So, all of the channels within the through path are switched correctly without altering the VCI value of cells on these channels.

Four parameters are needed to define a through path on a FORE switch fabric: input port, input VPI, output port, and output VPI. Through paths are represented as follows:

```
<iport> <ivpi> <oport> <ovpi>
```

The VCI value remains unchanged when cells are switched via a through path. For example, the through path A4|10 -> B4|20 maps cells received on port A4 with VPI: 10 and any VCI to port B4 with VPI: 20 and the same VCI.

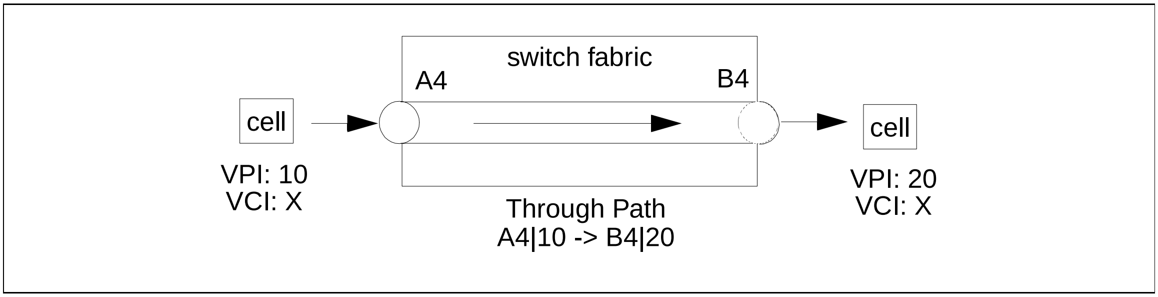

Figure 1.5 - An Example of a Through Path

By definition, through paths only switch cells in one direction; they are unidirectional. For example, switch fabric X is configured with the through path B1|20 -> C1|20. If a cell is received on port C1 with VPI: 20, it is not transmitted on port B1 with a new VPI: 20. In order for this to happen, the through path C1|20 -> B1|20 must exist as well. Since through paths are unidirectional, two through paths are necessary for bidirectional communication.

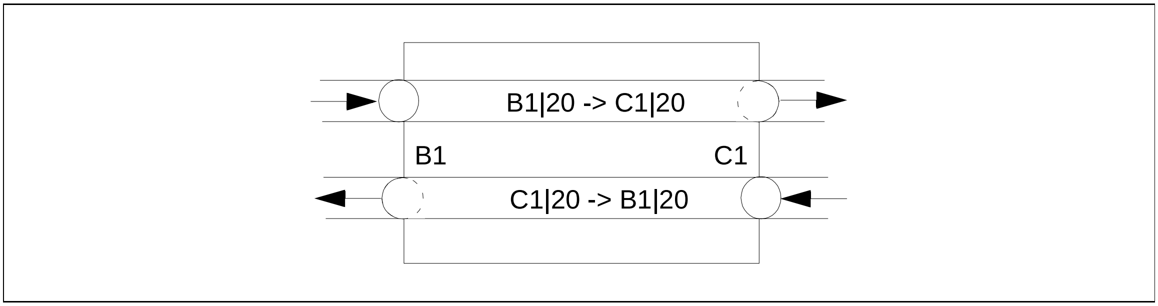

Figure 1.6 - Through Paths are Unidirectional

1.2.2 Originating and Terminating Paths

As previously noted, originating and terminating paths (also called virtual path terminators) are points at which a virtual path originates and terminates. For example, if a virtual path exists from switch fabric A to switch fabric B, then there must be an originating path on switch fabric A and a terminating path on switch fabric B.

An originating path is defined by two parameters: output VPI and output port. Similarly, a terminating path is defined by the parameters: input VPI and input port. Because originating and terminating paths do not define the way cells are switched through an ATM switch fabric, virtual channels must exist to switch cells from a terminating path to an originating path. (See the section about virtual channels for more information.) Originating and terminating paths are the endpoints of virtual paths and are used primarily for bandwidth allocation.

The bandwidth allocated to originating and terminating paths is used to control the amount of virtual channel (VCC) bandwidth entering or leaving a virtual path. The total guaranteed bandwidth used by virtual channels on an originating path or a terminating path cannot exceed the amount of bandwidth allocated to that path. For example, as illustrated in Figure 1.7, if each of the four virtual channels shown is using 10 Mbps of bandwidth, then the originating and terminating paths must have at least 40 Mbps of bandwidth allocated.

NOTE

UBR traffic bandwidth, which is a “best effort” service class, is not limited by the VP’s allocated bandwidth since its bandwidth is not guaranteed. Actual UBR VCC traffic transmitted within a VP may exceed the VP’s allocated bandwidth.

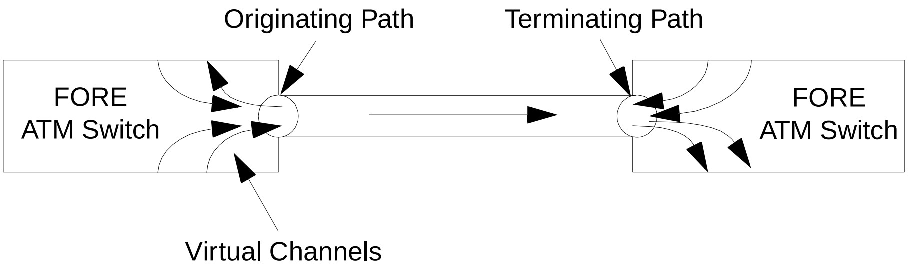

Figure 1.7 - Using Originating and Terminating Paths for Bandwidth Allocation

1.3 Listing Virtual Paths

1.3.1 Listing Through Paths

By logging in to AMI (see Chapter 1 of the *ATM Management Interface (AMI) Manual* for information about logging into AMI), it is possible to display either all of the existing through paths on an individual switch fabric or all of the existing through paths on a specified port. To list all of the existing through paths on an individual switch fabric, enter the following:

```
configuration vpc show
```

```
Input       Output
Port   VPI  Port   VPI  UPC  Prot   Name
3B1     40  3B4     40    0  pvc    customer_a
3B1     75  3B5     75    0  pvc    customer_b
3B2     95  3B3     95    0  pvc    customer_e
3B6     62  3B2     62    0  pvc    customer_c
3B6     68  3B3     68    0  pvc    customer_d
```

The fields in this display are defined as follows:

Field	Description
Input Port	The incoming port number of the through path.
Input VPI	The incoming virtual path number.
Output Port	The outgoing port number of the through path.
Output VPI	The outgoing virtual path number.
UPC	The integer index that refers to a specific UPC traffic contract assigned to this through path. UPC contracts can be displayed using **conf upc show**.
Prot	The type of protocol running on this channel.
Name	The user-assigned name which helps to identify this through path uniquely.

To list advanced options about all of the existing virtual (through) paths, enter the following parameters:

`configuration vpc show advanced`

```
Input      Output
Port  VPI  Port  VPI  Shape  ConType
3B1    40  3B4    40         N/A
3B1    75  3B5    75         N/A
3B2    95  3B3    95         tran-tran-pmp
3B6    62  3B2    62         tran-tran-pp
3B6    68  3B3    68         N/A
```

The fields in the advanced display are defined as follows:

Field	Description
Input Port	The incoming port number of the through path.
Input VPI	The incoming virtual path number.
Output Port	The outgoing port number of the through path.
Output VPI	The outgoing virtual path number.
Shape	Indicates whether or not traffic shaping has been enabled for this path. This field only applies to the Series C network modules.
ConType	The connection type for the endpoints of this path with respect to a particular network. **`Orig`** (originating) means that the ingress/egress endpoint of the path is connected to the source node which is outside the network, **`tran`** (transit) means that the ingress/egress endpoint of the path is connected to a node within the network, and **`term`** (terminating) means that the ingress/egress endpoint of the path is connected to the destination node which is outside the network. **`pp`** means this is labelled as a point-to-point path, **`pmp`** means this is labelled as a point-to-multipoint path, **`mpp`** means this is labelled as a multipoint-to-point path. **`mpmp`** means this is labelled as a multipoint-to-multipoint path.

1.3.2 Listing Originating and Terminating Paths

By logging in to AMI, it is possible to display either all of the existing originating and terminating paths on an individual switch fabric or on a specified port. To list all of the originating and terminating paths on an individual switch fabric, enter the following parameters:

```
configuration vpt show
```

```
Input      Output
Port  VPI  Port  VPI   ResBW  CurBW MinVCI MaxVCI  VCs  Protocol
1C1     0  terminate     N/A   0.8K      1    511    6  pvc
1C1     1  terminate    1.0M   0.8K      1    511    2  pvc
1C2     0  terminate     N/A   0.8K      1    511    6  pvc
1C3     0  terminate     N/A   0.8K      1    511    6  pvc
1C4     0  terminate     N/A   0.8K      1    511    6  pvc
1CTL    0  terminate     N/A   7.6K      1   1023   24  pvc
originate  1C1     0     N/A   0.8K      1    511    6  pvc
originate  1C1     1    1.0M   0.8K      1    511    2  pvc
originate  1C2     0     N/A   0.8K      1    511    6  pvc
originate  1C3     0     N/A   0.8K      1    511    6  pvc
originate  1C4     0     N/A   0.8K      1    511    6  pvc
originate  1CTL    0     N/A   7.6K      1   1023   30  pvc
```

The fields in this display are defined as follows:

Field	Description
Input Port	The incoming port number of the vpt. Shows **originate** if it is an originating path.
Input VPI	The incoming virtual path number.
Output Port	The outgoing port number of the vpt. Shows the number of the output port of the vpt. Shows **terminate** if it is a terminating path.
Output VPI	The outgoing virtual path number.
ResBW	The maximum amount of bandwidth, in Kbps, that is reserved for the virtual channels using this vpt. A value of **N/A** indicates that this path is an elastic path. Elastic paths allocate and de-allocate bandwidth for their channels from the link.
CurBW	The amount of bandwidth, in Kbps, being used by the virtual channels using this vpt.
MinVCI	The bottom number for the range of VCIs that are reserved for VCCs on this virtual path terminator. The default is **1**.
MaxVCI	The top number for the range of VCIs that are reserved for VCCs on this virtual path terminator. The default is **511**.
VCs	The number of virtual channels that are currently using this vpt.
Protocol	The type of protocol running on this channel.

To list all of the advanced options about all of the existing virtual path terminators, enter the following parameters:

```
configuration vpt show advanced
```

```
Input      Output
 Port  VPI  Port  VPI   ShapeVPI LoopVPI VBROB BuffOB
 1B1     0  terminate       N/A     N/A   N/A    N/A
 1B1     1  terminate       N/A     N/A   N/A    N/A
 originate  1B1     0                    port   port
 originate  1B1     1                10  port   port
```

The fields in the advanced display are defined as follows:

Field	Description
Input Port	The incoming port number of the vpt. Shows **originate** if it is an originating path.
Input VPI	The incoming virtual path number.
Output Port	The outgoing port number of the vpt. Shows **terminate** if it is a terminating path.
Output VPI	The outgoing virtual path number.
ShapeVPI	Shows the output port on which traffic shaping has been enabled for this originating vpt. This only applies to Series C network modules. **N/A** means that shaping has not been configured on this port.
LoopVPI	Shows the looping port that has been configured to loop traffic on that port for shaping on a Series D port. **N/A** means that looping has not been configured on this port.
VBROB	The bandwidth overbooking level assigned to this vpt, specified as a percentage. The default is 100, which means that no overbooking has been defined. Values less than 100 cause underbooking. Values greater than 100 denote overbooking. port means this is an elastic path. Since elastic paths derive their overbooking factors from their parent ports, use **conf port show** to display the overbooking value.
BuffOB	The buffer overbooking level assigned to this vpt, specified as a percentage. The default is 100, which means that no overbooking has been defined. Values less than 100 cause underbooking. Values greater than 100 denote overbooking. port means this is an elastic path. Since elastic paths derive their overbooking factors from their parent ports, use **conf port show** to display the overbooking value.

NOTE

For more information about setting overbooking parameters (VBROB and BuffOB), see Section 1.5.2.

1.4 Virtual Channels

Virtual channels "ride" inside of virtual paths. The combination of the two specifies a virtual connection. On a switch fabric, each virtual channel switches cells with a specific VPI and VCI received on a specific port to another port with a new VPI and a new VCI. Unlike through paths, which carry one or more VCCs, virtual channels describe a single virtual connection between two endpoints.

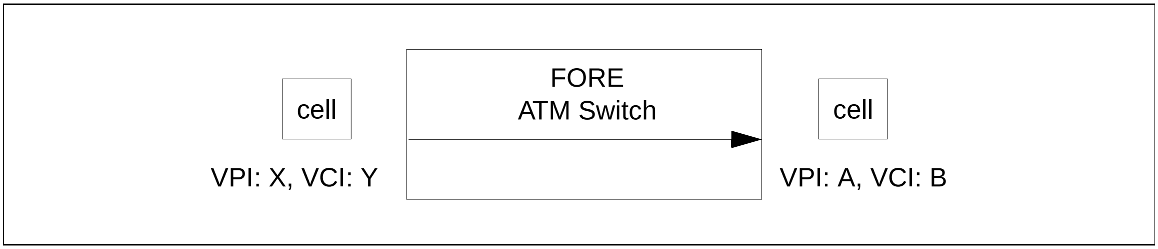

Figure 1.8 - An Example of a Virtual Channel

Six parameters are needed to define a virtual channel: input port, input VPI, input VCI, output port, output VPI, and output VCI. Virtual channels are represented by the following notation:

```
<iport> <ivpi> <ivci> <oport> <ovpi> <ovci>
```

Virtual channels switch cells using both the VPI and VCI values. Both the VPI and VCI values may change when a cell is switched via a virtual channel. For example, the virtual channel C2|1|20 -> D2|9|25 switches cells received on port C2 with VPI: 1 and VCI: 20 such that they are transmitted out port D2 with VPI: 9 and VCI: 25.

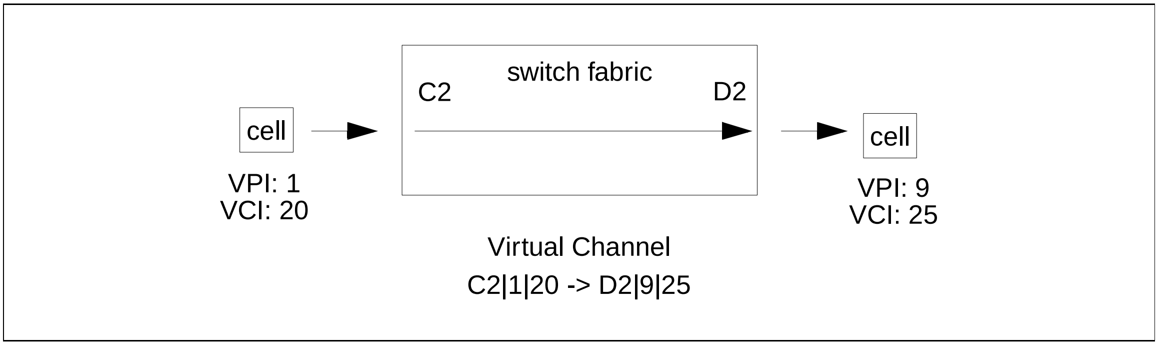

Figure 1.9 - Example of a Virtual Channel

In order to establish two-way communications between two ports on a switch fabric, two virtual channels are necessary because virtual channels are unidirectional. For example, switch fabric A is configured with the virtual channel C3|7|12 -> D1|8|2. If a cell is received on port D1 with VPI: 8 and VCI: 2, it is not transmitted out port C3 with VPI: 7 and VCI: 12. An additional channel, namely D1|8|2 -> C3|7|12, would have to exist.

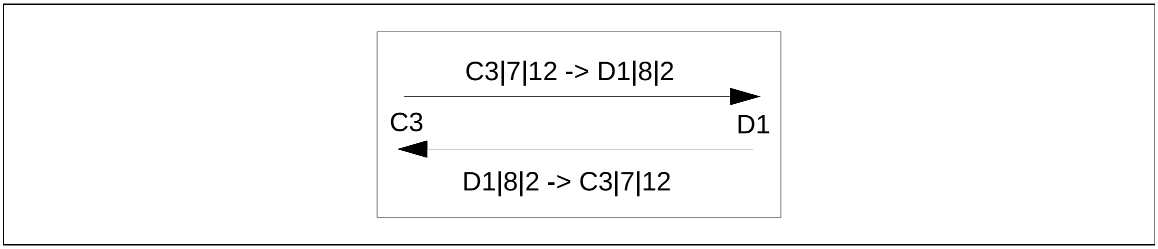

Figure 1.10 - Virtual Channels are Unidirectional

Before a virtual channel can be created, the corresponding terminating and originating paths must exist. For example, before the channels shown on the switch fabric in Figure 1.11 can be created, the terminating path C3|3 must exist.

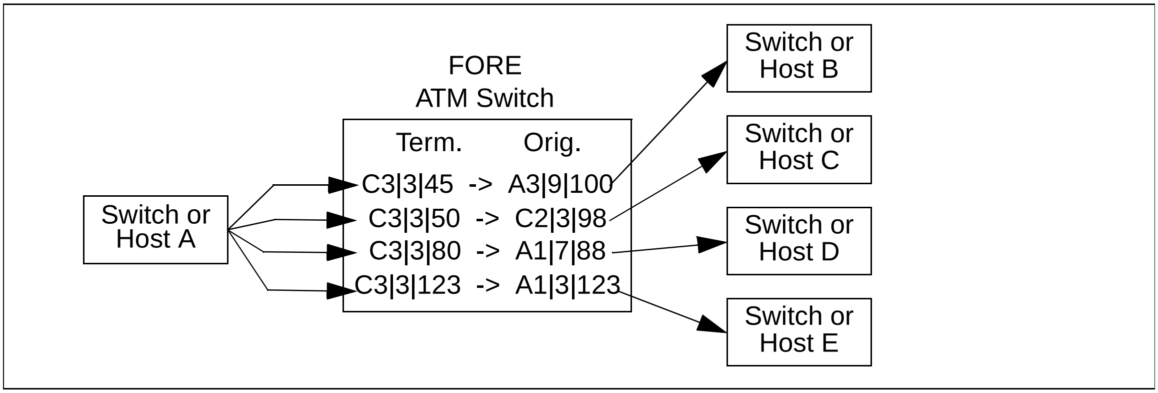

Figure 1.11 - Virtual Channels Created on Terminating Path C3|3

Similarly, before the virtual channels shown in Figure 1.12 can be created, the originating path C2|2 must exist.

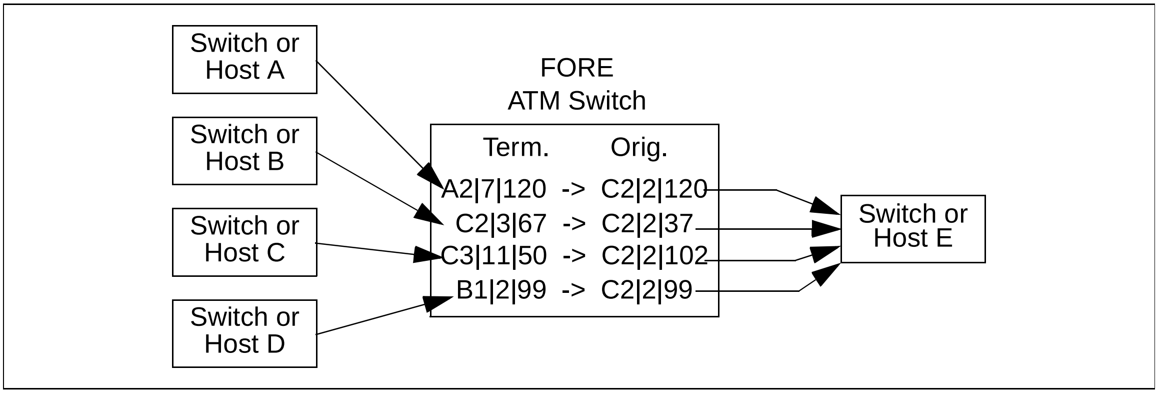

Figure 1.12 - Virtual Channels Created on Originating Path C2|2

Furthermore, in these examples, the terminating path C3|3 and originating path C2|2 must have enough bandwidth allocated to support the total bandwidth used by the virtual channels (see Figure 1.7).

1.4.1 Smart Permanent Virtual Circuits

Smart Permanent Virtual Circuits (SPVCs) are connections that go across multiple switch fabrics. An SPVC looks like a PVC at the local and remote endpoints with an SVC (Switched Virtual Circuit) in the middle. SVCs are channels established on demand by network signalling. Similar to a dialed telephone call, SVCs transport information between two locations and last only for the duration of the transfer.

SPVCs are more robust than PVCs. If a link carrying a PVC goes down, then the PVC goes down. If a link carrying a SPVC goes down and there is an alternate route, then the end switch fabrics of the SPVC automatically reroute the SPVC around the failed link.

As shown in Figure 1.13, interswitch links exist between the *FORE* switches. The endpoints exist at switch A and switch E. If a link goes down between switch A and switch B, an SPVC can reroute the cell (via an SVC) through switch C.

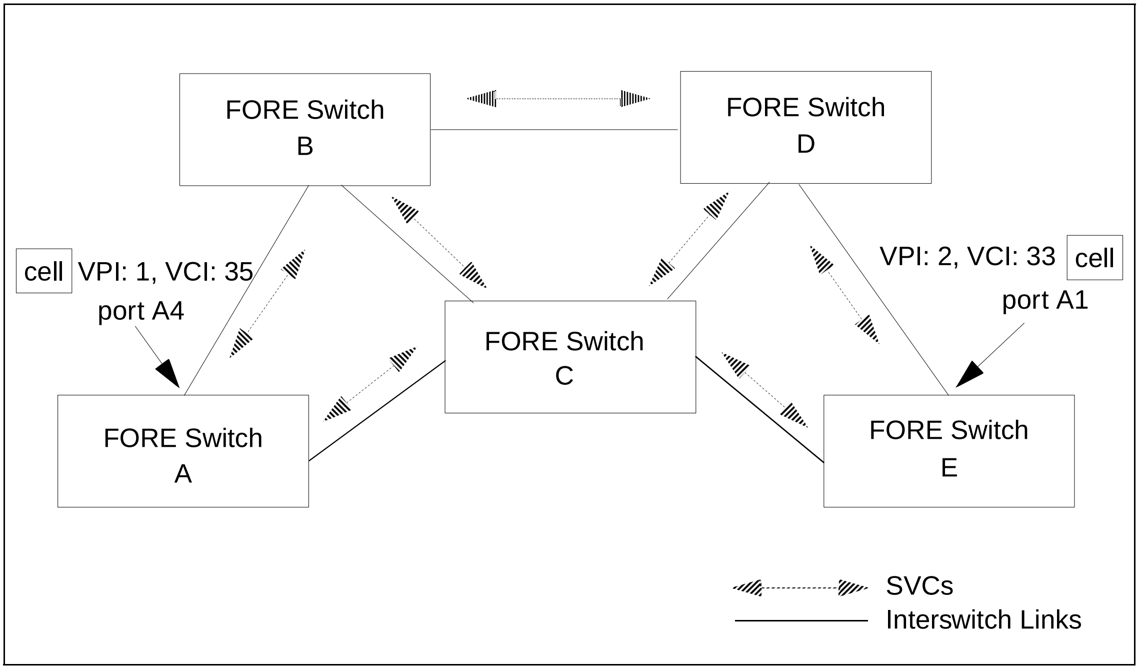

Figure 1.13 - The Path of a Cell Via SPVCs

1.4.2 Listing Virtual Channels

By logging in to AMI, you can display either all of the existing virtual channels on an individual switch fabric or on a specified port. To list all of the virtual channels on an individual switch fabric, enter the following parameters:

```
configuration vcc show
```

```
Input            Output
Port   VPI  VCI  Port   VPI  VCI  UPC  Protocol  Name
3B1      0    5  3CTL     0   49    0  fsig      N/A
3B1      0   14  3CTL     0   48    0  spans     N/A
3B1      0   15  3CTL     0   47       spans     N/A
3B1      0   16  3CTL     0   50       fsig      N/A
3B1      0  100  3B4      0  100    0  pvc       N/A
3B2      0    5  3CTL     0   53    0  fsig      N/A
3B2      0   14  3CTL     0   52    0  spans     N/A
3B2      0   15  3CTL     0   51       spans     N/A

Press return for more, q to quit: q
```

The fields in this display are defined as follows:

Field	Description
Input Port	The incoming port number of the virtual channel.
Input VPI	The incoming virtual path number.
Input VCI	The incoming virtual channel number.
Output Port	The outgoing port number of the virtual channel.
Output VPI	The outgoing virtual path number.
Output VCI	The outgoing virtual channel number.
UPC	The integer index that refers to the specific UPC traffic contract assigned to this VCI.
Protocol	Indicates what type of channel this is. Can be `spans`, `pvc`, `fsig`, `spvc`, or `rcc`. rcc is the routing control channel (0, 18) on PNNI links over which PNNI exchanges routing information. `fsig` stands for ATM Forum signalling.
Name	The unique, user-assigned name for this channel. If no name is assigned, shows `N/A`.

To list advanced information about all of the existing permanent virtual channels on a switch board, enter the following parameters:

`configuration vcc show advanced`

```
 Input        Output
 Port  VPI  VCI  Port  VPI  VCI  Protocol  ConType
 3B1     0    5  3CTL    0   49  uni       N/A
 3B1     0   14  3CTL    0   48  spans     N/A
 3B1     0   15  3CTL    0   47  spans     N/A
 3B1     0   16  3CTL    0   50  uni       N/A
 3B1     0  100  3B4     0  100  pvc       tran-tran-pp
 3B2     0    5  3CTL    0   53  uni       N/A
 3B2     0   14  3CTL    0   52  spans     N/A
 3B2     0   15  3CTL    0   51  spans     N/A
 3B2     0   16  3CTL    0   54  uni       N/A
Press return for more, q to quit: q
```

The fields in the advanced display are defined as follows:

Field	Description
Input Port	The incoming port number of the virtual channel.
Input VPI	The incoming virtual path number.
Input VCI	The incoming virtual channel number.
Output Port	The outgoing port number of the virtual channel.
Output VPI	The outgoing virtual path number.
Output VCI	The outgoing virtual channel number.
Protocol	Indicates what type of channel this is. Can be `spans`, `pvc`, `uni`, `spvc`, or `rcc`. rcc is the routing control channel (0, 18) on PNNI links over which PNNI exchanges routing information.
ConType	The connection type for the endpoints of this channel with respect to a particular network. **`Orig`** (originating) means that the ingress/egress endpoint of the channel is connected to the source node which is outside the network, **`tran`** (transit) means that the ingress/egress endpoint of the channel is connected to a node within the network, and **`term`** (terminating) means that the ingress/egress endpoint of the channel is connected to the destination node which is outside the network. **`pp`** means this is labelled as a point-to-point channel, **`pmp`** means this is labelled as a point-to-multipoint channel, **`mpp`** means this is labelled as a multipoint-to-point channel. **`mpmp`** means this is labelled as a multipoint-to-multipoint channel.

1.5 Creating PVCs and SPVCs

FORE’s ATM network modules provide ATM transmission connectivity, while its intelligent network modules, such as the Circuit Emulation Services (CES) network module, provide adaptation for ports carrying one transmission format (e.g., TDM) to ATM cells. This section describes how to create the following from one ATM port to another:

- a permanent virtual path (through path)
- a permanent virtual path terminator (originating or terminating path)
- a permanent virtual channel through the network
- a smart permanent virtual channel through the network

This section assumes that the physical port parameters of the switches have already been configured and that the ATM network module traffic models have been set appropriately. For more information about these configurations, see Part 2 of the *AMI Configuration Commands Reference Manual.*

NOTE

When these paths and channels are created, a command is entered automatically into the current configuration database (CDB), meaning that these paths and channels are created every time the switch control processor (SCP) is restarted. The CDB should be backed up frequently.

1.5.1 Creating a Through Path

To create a new through path, log in to AMI and enter the following parameters:

```
conf vpc new <iport> <ivpi> <oport> <ovpi> [-upc <index>]
                    [-name <name>]
```

The optional parameters for call records and VP shaping (using Series C network modules) are as follows:

```
[-inctype (orig | tran | term) -outctype (orig | tran | term)
           [pmp |mpp | mpmp]]\-shapeivpi <vpi>]
```

These parameters are defined as follows:

Parameter	Description
iport	The incoming port number.
ivpi	The incoming virtual path number.
oport	The outgoing port number.
ovpi	The outgoing virtual path number.
-upc <index>	The integer index that refers to a specific UPC traffic contract. If no index is specified, then no traffic policing will take place on this VPI. It is assigned a UPC index of 0, and all traffic on this VPI is treated as UBR traffic. This is the default.
-name <name>	The name you want to assign to this through path to help identify it uniquely. It is most useful for billing purposes so you can identify which paths are being used by which customers. Can be up to 32 ASCII characters long.
-inctype (orig | tran |term)	The path connection type for the incoming path. For billing purposes, it denotes on which switch this path is arriving. **Orig** (originating) means that the ingress endpoint of the path is connected to the source node which is outside the network, **tran** (transit) means that the ingress endpoint of the path is connected to a node within the network, and **term** (terminating) means that the ingress endpoint of the path is connected to the destination node which is outside the network.
-outctype(orig | tran |term)	The path connection type for the outgoing path. For billing purposes, it denotes on which switch this path is leaving. **Orig** (originating) means that the egress endpoint of the path is connected to the source node which is outside the network, **tran** (transit) means that the egress endpoint of the path is connected to a node within the network, and **term** (terminating) means that the egress endpoint of the path is connected to the destination node which is outside the network.
pmp[1]	Indicates this is a point-to-multipoint path.
mpp	Indicates this is a multipoint-to-point path.
mpmp	Indicates this is a multipoint-to-multipoint path.

Parameter	Description
-shapeivpi <vpi>[2]	The incoming VPI for this through path. When the traffic shaping port is not the port connected to the WAN, a through path must be created from the WAN port to the traffic shaping port. Cells arrive from the network at the traffic shaping port with this value equal to the VPI of the terminating path at the traffic shaping port. This parameter only applies to the Series C network modules.

[1] By indicating **pmp**, **mpp**, or **mpmp**, you are only assigning a label for record keeping purposes. The switch does not necessarily create the type of path you have specified. If you assign a connection type, but do not assign a **pmp**, **mpp**, or **mpmp** label, the switch assigns a label of **pp** (point-to-point).

[2] If you want to shape traffic on more than two ports on a given Series C network module, it is recommended that you set the traffic memory model to model number 5 for that network module using **conf module traffic c setmodel**.

NOTE

Terminating and originating paths cannot be created across the intra-fabric ports on an *ASX*-1000, *ASX*-1200, or *TNX*-1100; only through paths can be created across the intra-fabric ports as shown in the third example.

The following is an example of how to create a virtual path which specifies a name:

```
myswitch::configuration vpc> new 3b1 75 3b5 75 -name customer_b
```

The following is an example of how to create a virtual path which specifies a name and a connection type:

```
myswitch::configuration vpc> new 3b6 62 3b2 62 -name customer_c -inctype tran
-outctype tran
```

The following is an example of how to create a virtual path on an *ASX*-1000, *ASX*-1200, or *TNX*-1100. To create a through path going in port 2A1, VPI 1 on the switch board installed in slot 2 and going out port 4B1, VPI 1 on the switch board installed in slot 4, enter the following:

```
myswitch::configuration vpc> new 2a1 1 2e4 1
myswitch::configuration vpc> new 2e4 1 2a1 1

myswitch::configuration vpc> new 4b1 1 4e2 1
myswitch::configuration vpc> new 4e2 1 4b1 1
```

In the first line in the first pair, notice that the output port is 2E4. This is the intra-fabric port. The 2 means the connection is coming out of the switch board in slot 2 through the intra-fabric port. The E represents the intra-fabric port. The 4 means the connection is destined for switch board in slot 4. 2E4 then becomes the input port in the second line.

In the first line in the second pair, notice that the output port is 4E2. This is the intra-fabric port. The 4 means the connection is coming out of the switch board in slot 4 through the intra-fabric port. The E represents the intra-fabric port. The 2 means the connection is destined for switch board in slot 2. 4E2 then becomes the input port in the second line.

At the same time, a command is entered automatically into the current configuration database, which means that this virtual path is created every time the SCP is restarted.

1.5.2 Creating an Originating or Terminating Path

To create an originating or terminating path (virtual path terminator), log in to AMI and enter the following parameters:

```
configuration vpt new <port> <vpi> (term | orig) [-reserved <Kbs>]\
                  [-minvci <vci>] [-maxvci <vci>]
```

The **-reserved**, **-minvci**, and **-maxvci** parameters are optional for originating or terminating paths. The advanced traffic management options for originating paths are as follows:

```
                  [-shapeovpi <vpi>] [-loopvpi <vpi>]
               [-vbrob <percent>] [-vbrbuffob <percent>]
```

The advanced QoS options for creating originating or terminating paths are as follows:

```
                  [-cbr (none | default | <qosindex>)]
                 [-rtvbr (none | default | <qosindex>)]
                [-nrtvbr (none | default | <qosindex>)]
                  [-ubr (none | default | <qosindex>)]
                  [-abr (none | default | <qosindex>)]
```

NOTE

The *<qosindex>* must exist (**conf qosext new**) before it can be applied to the originating/terminating path.

These parameters are defined as follows:

Parameter	Description
port	The port number for this vpt.
vpi	The path number for this vpt.
term	Specifies that the vpt to be created is a terminating path.
orig	Specifies that the vpt to be created is an originating path.
reserved	The amount of bandwidth, in Kbps, that you want to reserve on this vpt. If this option is not used, an elastic path is created. Elastic paths allocate and de-allocate bandwidth for their channels from the link.
minvci	The bottom number of the range of VCIs to be reserved for VCCs on this vpt. The default is **1**.
maxvci[1]	The top number of the range of VCIs to be reserved for VCCs on this vpt. The default is **511**.

Parameter	Description
shapeovpi	The output path on a traffic shaping originating vpt. Setting this value configures traffic shaping on the originating path. Cells bound for the network leave the traffic shaping port with this VPI. When the traffic shaping port is the WAN port, this value equals the input VPI of the originating path. If the traffic shaping port is not the WAN port, this value equals the input VPI of the through path from the shaping port to the WAN port. This parameter only applies to the Series C network modules. See Section 1.5.2.1.
loopvpi	The originating vpi will be shaped by a through path going to a Series D network module. You should enter the input vpi of the through path that goes from the looping port to the WAN port. This option is also used when creating the through path that connects from the WAN port to the looping port. The through path loopvpi should be the same vpi as the terminating path on the looping port. See Section 1.5.2.1 for more information.
vbrob	The bandwidth overbooking level for this vpt, specified as a percentage. Valid values are integers from 1 to 32,767. **100** means that no overbooking has been defined. Values less than **100** cause underbooking. Values greater than **100** cause overbooking. Overbooking cannot be specified on an elastic path. Therefore, you can only specify an overbooking factor for an originating path when you also have reserved bandwidth for the path (i.e., specified the **-reserved** <*Kbs*> parameter).
vbrbuffob	The buffer overbooking level for this vpt, specified as a percentage. Valid values are integers greater than or equal to 1. **100** means that no overbooking has been defined. Values less than **100** cause underbooking. Values greater than **100** cause overbooking. Overbooking cannot be specified on an elastic path. Therefore, you can only specify an overbooking factor for an originating path when you also have reserved bandwidth for the path (i.e., specified the **-reserved** <*Kbs*> parameter).
none	The specified class of service is not supported.
default	The default parameters of 0 CTD, 0 CDV, and 0 CLR are to be used for the CBR class of service.
qosindex	The index of the set of QoS extension parameters. See **conf qosext show** for this number.

[1] In the **modify** command, this is the maximum VCI number to be supported on the control port. The allowable range is from **0** - **8191** on the *ASX*-4000 and the *ESX*-3000. The allowable range is **0** - **4095** on all other switch platforms. The default is **4095** on the *ASX*-4000 and the *ESX*-3000. The default is **1023** on all other switch platforms.

NOTE

Bandwidth and/or buffer overbooking are used when the number of VBR VCCs configured across an originating or terminating VPT exceeds the guaranteed capacity of the VP. While overbooking removes service guarantees (in the case when all users transmit at full contract rate simultaneously), a network can be engineered in a more cost-effective manner if it is assumed that all VCC sources do not transmit simultaneously.

Guideline: An initial buffer overbooking value of 500-700% is recommended until live traffic patterns and VPC utilization can be measured. If bandwidth is still under-utilized, a higher VBROB setting can be used.

The following is an example of how to create a terminating path:

```
myswitch::configuration vpt> new 3b3 99 term
Would you like to create the originating side also [y]? y
```

The following is an example of how to create a originating path:

```
myswitch::configuration vpt> new 3b4 88 orig
Would you like to create the terminating side also [y]? y
```

The following is an example of how to delete a terminating path:

```
myswitch::configuration vpt> del 3b4 88 term
Would you like to delete the originating side also [y]? y
```

The following is an example of how to delete an originating path:

```
myswitch::configuration vpt> del 3b3 99 orig
Would you like to delete the terminating side also [y]? y
```

If you do not specify **`term`** or **`orig`**, the switch automatically deletes both sides of the path:

```
myswitch::configuration vpt> del 3b4 88
```

NOTE

Before deleting a virtual path, you must first delete all VCCs which use that path.

1.5.2.1 Shaping Multiple Originating Paths on a Single Port

This feature allows you to shape several originating paths to be output on a single port. This feature is useful if you have several remote sites interconnected by a PVP mesh. If you only need to shape one originating path, you can simply use the **`conf port traffic d ratelimit`** command on a Series D network module.

This feature requires the use of two ports. One port, which can be on a Series C, Series LC, or Series D network module, is used for originating and terminating path(s). This port is put in either physical or diagnostic loopback and is configured with the new option **`-loopvpi`** under **`conf vpt new`**. The other port provides the actual output on a shaped through path, so this port must be on a Series D network module.

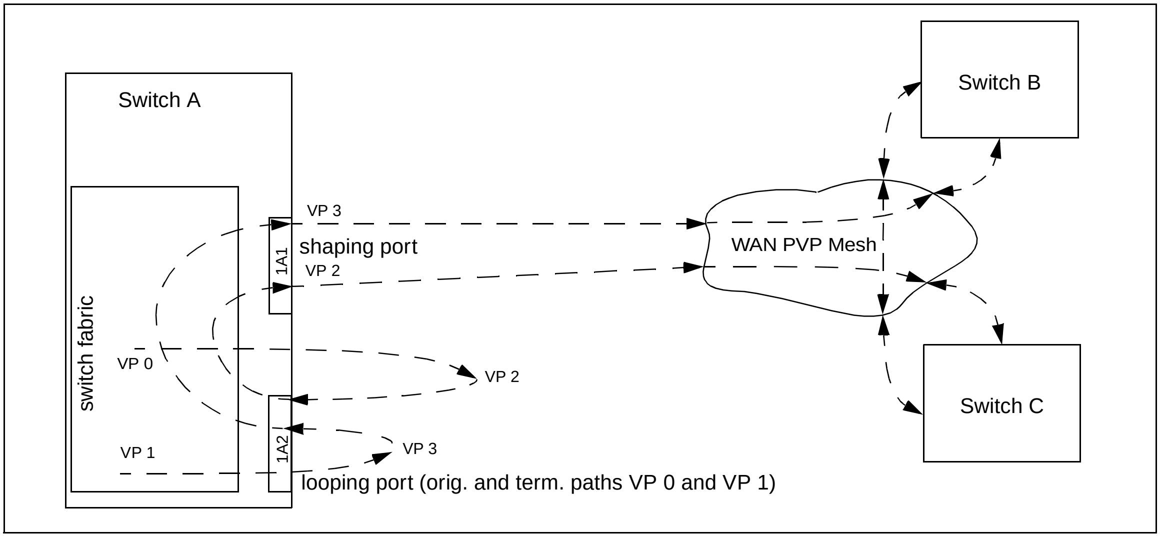

Figure 1.14 - PVPs Looped through Port 1A2 and Output on Port 1A1 to WAN

As shown in the example in Figure 1.14, the traffic that is going out originating path 0 on port 1A2 gets looped so that it is output to the WAN on through path 2 on shaping port 1A1. Similarly, the traffic that is going out originating path 1 on port 1A2 gets looped so that it is output to the WAN on through path 3 on shaping port 1A1.

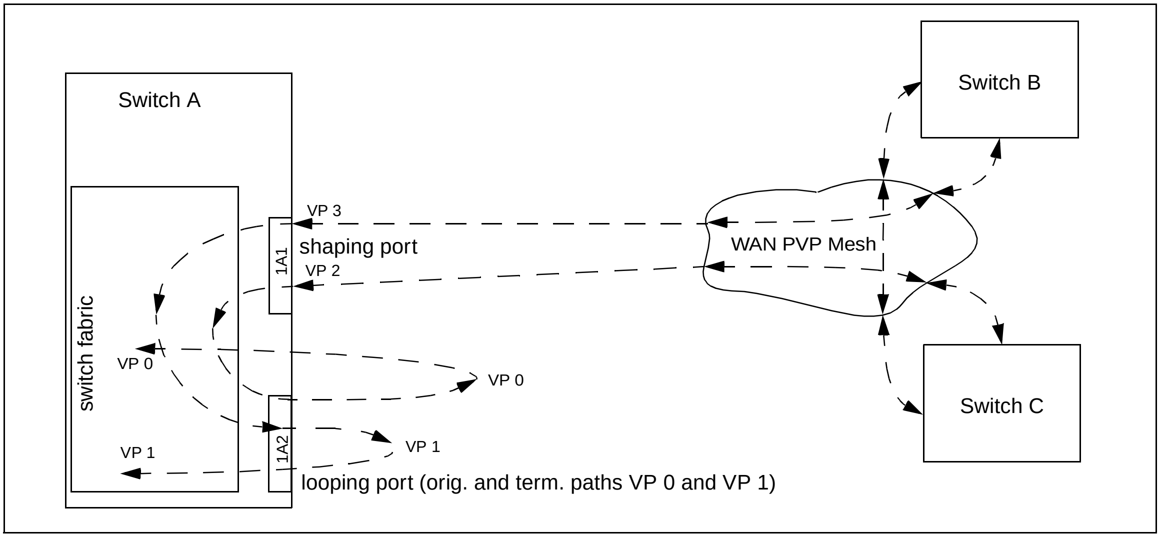

Figure 1.15 - PVPs Coming in Port 1A1 from WAN and Looped through Port 1A2

Then, as shown in the example in Figure 1.15, the traffic that is coming in through path 2 on port 1A1 from the WAN gets looped back to terminating path 0 on on port 1A2. Similarly, the traffic that is coming in through path 3 on port 1A1 from the WAN gets looped back to terminating path 1 on port 1A2.

In your own network, you will repeat this process on these same two ports for as many paths as you want to shape.

NOTE

Before you configure the paths, you should configure the network module that will contain the looping port to maximize the number of multicast connections. On a Series C network module, use model number 5 under **`conf module traffic c setmodel`**. On a Series LC network module, use model number 6, 7, or 8 under **`conf module traffic lc setmodel`**.

To configure the shaped originating paths as shown in the example in Figure 1.14 and Figure 1.15, perform the following steps:

1. First, create the UPC contract to be used on the through paths that are going to be output on the shaping port on the Series D network module to the WAN.

```
myswitch::configuration upc> new 2 cbr 42452 noGCRA -scheduling smoothed
```

NOTE

You must use the **-noGCRA** option to disable policing. This allow bursts without dropping cells. The shaped rate is equal to PCR for CBR traffic and equal to SCR for VBR traffic. You must also use the **smoothed** scheduling option so that a rate group is created when a PVC or PVP is configured with this UPC.

2. Next, delete any existing path(s) on the looping port as follows:

```
myswitch::configuration upc> conf vpt
myswitch::configuration vpt> conf vpt del 1a2 0
```

NOTE

By default, path 0 is the only path that exists. If path 1 already exists, you must delete it using the command **conf vpt del 1a2 1**.

3. Recreate the paths on the looping port using the **-loopvpi** option to ensure that cells from the WAN port get looped back into the terminating path.

```
myswitch::configuration vpt> new 1a2 0 -loopvpi 2
myswitch::configuration vpt> new 1a2 1 -loopvpi 3
```

4. Create a PVP to the shaping port on a Series D network module and apply the UPC contract that you created in step 1 so that it shapes the cells.

```
myswitch::configuration vpt> conf vpc
myswitch::configuration vpc> new 1a2 2 1a1 2 -upc 2
myswitch::configuration vpc> new 1a2 3 1a1 3 -upc 2
```

5. Create a PVP in the opposite direction and use **-loopvpi** to ensure that the cells on each through path get looped back the terminating paths on the looping port (1a2).

```
myswitch::configuration vpc> new 1a1 2 1a2 2 -loopvpi 0
myswitch::configuration vpc> new 1a1 3 1a2 3 -loopvpi 1
```

6. Now use a loopback command to loop the transmit side to the receive side on the looping port (1a2).

```
myswitch::configuration vpt> conf port admin 1a2 down
myswitch::configuration vpt> conf port sonet loop 1a2 diag
```

NOTE

Any required signalling paths must be recreated at this point.

7. Additional shaped originating paths can be added later on these same two ports as needed by repeating the steps in this section.

1.5.2.2 Terminating a PVC at a Switch

Sometimes it is necessary to create a PVC between a host and a switch fabric that is at a remote location. In this case, the PVC should be created from the host to the control port (CTL) of the switch and vice versa.

Some additional configuration is necessary for communication to be established between the host and the switch fabric. The switch needs an entry in its ATM ARP cache in order to send cells destined for the host with the correct VPI and VCI and to pass received cells with a specific VPI and VCI to IP. This configuration can be done using AMI as shown in the following subsections. (See the `atmarp (8c)` man page for more information.)

1.5.2.3 Creating ATM ARP Entries

To create a FORE IP PVC ARP entry, log in to AMI. Data on this PVC is encapsulated using null encapsulation (also known as VC-based multiplexing) as specified in RFC-1483. Enter the following parameters:

```
configuration atmarp newforeip <host> <vpi> <vci> (4|5) [<interface>]
```

These parameters are defined as follows:

Parameter	Description
host	The IP address of the remote host.
vpi	The virtual path number of the FORE IP PVC. Must be 0.
vci	The virtual channel number of the FORE IP PVC.
4 | 5	The connection's ATM Adaptation Layer (AAL) type. The default is 4.
interface	The FORE IP interface to be used for this connection. The default is asx0.

Once the parameters are entered, the entry is created instantly by the SCP. At the same time, a command is entered automatically into the current CDB which creates this ATM ARP entry each time the SCP is restarted.

To create a new Classical IP PVC ARP entry, log in to AMI. All data is sent LLC/SNAP encapsulated. Enter the following parameters:

```
configuration atmarp newclassicalip <host> <vpi> <vci> [<interface>]
```

These parameters are defined as follows:

Parameter	Description
host	The host IP address of the remote IP endstation.
vpi	The virtual path number of the Classical IP PVC.
vci	The virtual channel number of the Classical IP PVC.
interface	The Classical IP interface to be used for this connection: qaa0, qaa1, qaa2, or qaa3. The default is qaa0.

Once the parameters are entered, the Classical IP PVC ARP entry is created instantly by the SCP. At the same time, a command is entered automatically into the current CDB which creates this ATM ARP entry each time that the SCP is restarted.

1.5.2.4 Listing ATM ARP Entries

To verify that the ARP entries exist correctly for the outgoing PVC connection from the SCP to the host, display the ATM ARP cache by logging in to AMI and entering the following parameters:

```
configuration atmarp show
```

```
IPaddress        If    VPI    VCI    AAL    Type              Direction
198.29.22.9      asx0  0      63     aal5   foreIpSVC         pending
198.29.22.15     asx0  0      231    aal5   foreIpSVC         pending
198.29.22.37     asx0  0      65     aal34  foreIpSVC         pending
IPaddress        If    NSAP Address
198.29.17.3      qaa0  0x47.0005.80.ffe100.0000.f21b.0138.002048102754.00
198.29.17.10     qaa0  0x47.0005.80.ffe100.0000.f21b.0137.002048100be6.00
198.29.17.15     qaa0  0x47.0005.80.ffe100.0000.f21b.0137.00204810048d.00
198.29.17.52     qaa0  0x47.0005.80.ffe100.0000.f21b.0138.0020481b0138.00
```

The fields in this display are defined as follows:

Field	Description
IPaddress	The IP address for this connection.
If	The name of the IP interface for this connection.
VPI	The virtual path number.
VCI	The virtual channel number.
AAL	The AAL type of the given connection.
Type	Shows what kind of connection this is. Can be `foreIpPVC`, `foreIpSVC`, `classicalIpPVC`, or `classicalIpSVC`.
Direction	`Outgoing` means this is an outgoing connection. `Incoming` means this is an incoming connection. `Pending` means that a connection has not (yet) been established. `Incomplete` means that the IP-to-ATM address mapping is not yet known for the given IP address.
NSAP Address	The NSAP address for this connection.

1.5.3 Creating a Virtual Channel

To create a new virtual channel, log in to AMI and enter the following parameters:

```
configuration vcc new <iport> <ivpi> <ivci> <oport> <ovpi> <ovci>
                [-upc <index>] [-name <name>]
```

The advanced options for call records are as follows:

```
-inctype (orig | tran | term) -outctype (orig | tran | term)
                [pmp | mpp | mpmp]]
```

These parameters are defined as follows:

Parameter	Description
iport	The incoming port number.
ivpi	The incoming virtual path number.
ivci	The incoming virtual channel number.
oport	The outgoing port number.
ovpi	The outgoing virtual path number.
ovci	The outgoing virtual channel number.
-upc <index>	The integer index that refers to a specific UPC traffic contract. If no index is specified, then no traffic policing will take place on this VCI. It is assigned a UPC index of 0, and all traffic on this VCI is treated as UBR traffic. This is the default.
name	The name you want to assign to this channel to identify it uniquely. It is useful for billing purposes so you can identify which channels are being used by which customers. Can be up to 32 ASCII characters long.
inctype	The channel connection type for the incoming channel. For billing purposes, it denotes on which switch this channel is arriving. **Orig** (originating) means that the ingress endpoint of the channel is connected to the source node which is outside the network, **tran** (transit) means that the ingress endpoint of the channel is connected to a node within the network, and **term** (terminating) means that the ingress endpoint of the channel is connected to the destination node which is outside the network.
outctype	The channel connection type for the outgoing channel. For billing purposes, it denotes on which switch this channel is leaving. **Orig** (originating) means that the egress endpoint of the channel is connected to the source node which is outside the network, **tran** (transit) means that the egress endpoint of the channel is connected to a node within the network, and **term** (terminating) means that the egress endpoint of the channel is connected to the destination node which is outside the network.
pmp[1]	Indicates this is a point-to-multipoint channel.
mpp	Indicates this is a multipoint-to-point channel.

Parameter	Description
mpmp	Indicates this is a multipoint-to-multipoint channel.

[1] By indicating `pmp`, `mpp`, or `mpmp`, you are only assigning a label for record keeping purposes. The switch does not necessarily create the type of channel you have specified. If you assign a connection type, but do not assign a `pmp`, `mpp`, or `mpmp` label, the switch assigns a label of `pp` (point-to-point).

The following is an example of how to create a uni-directional virtual channel which specifies the call record connection type as `transit`:

```
localhost::config vcc> new 3b1 0 100 3b4 0 100 -inctype tran -outctype tran
```

The following is an example of how to create a uni-directional virtual channel which has the name “customer_a” assigned to it:

```
myswitch::configuration vcc> new 3b2 0 145 3b3 0 145 -name customer_a
```

The following is an example of how to create a bi-directional virtual channel on an *ASX*-1000, *ASX*-1200, or *TNX*-1100. To create a VCC going in port 2A1, VPI 0, VCI 100 on the switch board installed in slot 2 and going out port 4B1, VPI 0, VCI 100 on the switch board installed in slot 4, enter the following:

```
myswitch::configuration vcc> new 2a1 0 100 2e4 0 100
myswitch::configuration vcc> new 2e4 0 100 2a1 0 100

myswitch::configuration vcc> new 4b1 0 100 4e2 0 100
myswitch::configuration vcc> new 4e2 0 100 4b1 0 100
```

In the first line in the first pair, notice that the output port is 2E4. This is the intra-fabric port. The 2 means the connection is coming out of the switch board in slot 2 through the intra-fabric port. The E represents the intra-fabric port. The 4 means the connection is destined for switch board in slot 4. 2E4 then becomes the input port in the second line.

In the first line in the second pair, notice that the output port is 4E2. This is the intra-fabric port. The 4 means the connection is coming out of the switch board in slot 4 through the intra-fabric port. The E represents the intra-fabric port. The 2 means the connection is destined for switch board in slot 2. 4E2 then becomes the input port in the second line.

Once the parameters are entered, the virtual channel is created instantly by the SCP. At the same time, a command is entered automatically into the current configuration database, which means that this virtual channel is created every time the SCP is restarted.

1.5.4 Creating SPVCs

A Smart Permanent Virtual Circuit Connection (SPVCC, hereafter referred to as SPVC) is a connection that goes across multiple switch fabrics. An SPVC looks like a PVC at the local and remote endpoints with an SVC in the middle. SPVCs are more robust than PVCs. If a link carrying a PVC goes down, then the PVC goes down. If a link carrying an SPVC goes down and there is an alternate route, then the end switch fabrics of the SPVC automatically reroute the SPVC around the failed link.

1.5.4.1 Creating a SPANS SPVC

To create a SPANS SPVC, you must configure both ends of the connection concurrently on the two switch fabrics. This means you must have an AMI session open on both the local switch fabric and the destination switch fabric. To create a new SPANS SPVC, log in to AMI and enter the following parameters:

```
configuration spvx spvcc spans new <port> <vpi> <vci> <dest-session>
        <dest-port> <dest-vpi> <dest-vci>\[-peak <Kb/sec>]
              [(source | destination | bidirectional)]
```

These parameters are defined as follows:

Parameter	Description
port	The port number on the local switch fabric.
vpi	The virtual path number on the local switch fabric.
vci	The virtual channel number on the local switch fabric.
dest-session	The IP address of the remote switch.
dest-port	The port number on the remote switch fabric.
dest-vpi	The virtual path number on the remote switch fabric.
dest-vci	The virtual channel number on the remote switch fabric.
-peak <Kb/sec>	The amount of peak bandwidth allocated for this SPANS SPVC, specified in kilobits per second. The default is **0**.
source | destination | bidirectional	**source** means a unidirectional SPANS SPVC going from the local switch fabric to the remote switch fabric will be created. **destination** means a unidirectional SPANS SPVC going from the remote switch fabric to the local switch fabric will be created. **bidirectional** means the pair of unidirectional SPANS SPVCs will be created. The default direction, if you do not specify one, is **bidirectional**.

NOTE

To create a bidirectional SPANS SPVC, you must either specify bidirectional, or you must set up two unidirectional SPANS SPVCs with one going in each direction.

To create a SPANS SPVC, you need to configure the two ends concurrently on the two switch fabrics. Therefore, you first need to open an AMI session to the destination switch fabric by using the SCP's IP address, along with the SNMP read-write community string. The following example depicts how to create a bidirectional SPVC from the local switch fabric (myswitch) to a remote switch fabric (198.29.22.46 named fishtank). The asterisk (*) in front of the prompt indicates that it is a remote session. To return to the local session, you must type **localhost** (instead of the prompt name).

```
myswitch::> open 198.29.22.46 private

Opening a session for “198.29.22.46”, please wait...
Connected to “198.29.22.46” (asx200bx).
*fishtank::> localhost

myswitch::> configuration spvx spvcc spans new <port> <vpi> <vci> <dest-session>
<dest-port> <dest-vpi> <dest-vci> \
[-peak <Kb/sec>] [(source | destination | bidirectional)]

myswitch::configuration spvx spvcc spans new 1c1 0 49 198.29.22.46 1b1 0 50
```

1.5.4.2 Creating a PP PNNI SPVC

To create a point-to-point (PP) PNNI SPVC, log in to AMI and enter the following parameters:

NOTE

You do not need to open a session to the destination switch to create a PP PNNI SPVC because all PNNI SPVCs are bidirectional.

```
conf spvx spvcc pnni pp new <port> <vpi> <vci> <destnsap>\
                [-priority <priority>]\
                  [-spvcid <index>]\
                  [-domainid <id>]\
        [-name <name>] [-destvpi <vpi> -destvci <vci>]\
              [-fupc <index>] [-bupc <index>]\
               [-reroute (enable|disable)]\
```

The advanced options are as follows:

```
               [-autoDtl (enable|disable)]\
              [-backoff (enable | disable)]\
                   [-dtlTag <tag>]\
         [-bearerClass (X | A | C)] [-clip <no|yes>]\
                [-QoSExpIndex <index>]\
     [-fqos (class0 | class1 | class2 | class3 | class4)]\
     [-bqos (class0 | class1 | class2 | class3 | class4)]
```

These parameters are defined as follows:

Parameter	Description
SPVC ID or <spvcid> or -spvcid <index>	The index number assigned to this point-to-point SPVC to identify it uniquely. For PNNI SPVCs, the index number may be assigned either by AMI or by the user. Valid values are integers between 1 and 65535.
port	The port number where the permanent portion of the SPVC resides on the source (local) switch fabric.
vpi	The virtual path number for the permanent portion of the SPVC on the source (local) switch fabric.
vci	The virtual channel number for the permanent portion of the SPVC on the source (local) switch fabric.
destnsap	The ATM NSAP address of the destination (remote) switch for this SPVC. Use the **conf spvx spvcc pnni destnsap** command on the destination switch to obtain this address. See Section 12.1.2.1 in Part 2 of the *AMI Configuration Commands Reference Manual* for more information.

Parameter	Description
-priority <priority>	The order in which and the frequency with which you want this SPVC to be tried for a connection. Higher priority SPVCs are tried more often than lower priority SPVCs. The value of **lowestpriority** determines the number of times each SPVC is tried before attempting a lower priority SPVC. Valid values are from **1** to **lowestpriority**, with 1 being the highest priority (the first and the most often tried) and with **lowestpriority** being the value configured under **conf spvx spvcc pnni parameters priority lowestpriority**. The default value is half of the value of **lowestpriority**, rounded off to the lowest integer. The value for **lowestpriority** is defaulted to 10. Point-to-point and point-to-multipoint SPVC priorities are shared meaning some point-to-point or PMP SPVCs can be assigned a higher priority compared to other point-to-point or PMP SPVCs. For point-to-point and PMP SPVCs with equal priority, point-to-point SPVCs have higher priority over PMP SPVCs.
-domainid <id>	The PNNI domain ID number on the source (local) switch fabric. The default is 1, which is the ID of the default routing domain.
-name <name>	An optional name for the user to assign to this PNNI SPVC to help uniquely identify it.
-destvpi <vpi>	The virtual path number on the destination switch fabric for this PNNI SPVC. When using this option, both the **-destvpi** and **-destvci** must be specified. If this option is not used, the destination switch uses any available value.
-destvci <vci>	The virtual channel number on the destination switch fabric for this PNNI SPVC. When using this option, both the **-destvpi** and **-destvci** must be specified. If this option is not used, the destination switch uses any available value.
-fupc <index>[1]	The forward (going from the local switch fabric to the remote switch fabric) UPC contract index assigned to this SPVC. To find the index you want, use the **conf upc show** command. If no index is specified, the default index of 0 (UBR best effort) is used.
-bupc <index>	The backward (going from the remote switch fabric to the local switch fabric) UPC contract index assigned to this SPVC. To find the index you want, use the **conf upc show** command. If no index is specified, the default index of 0 (UBR best effort) is used.
-reroute (enable | disable)	**enable** indicates that this SPVC will be examined periodically to see if it is using a sub-optimal route. If it is, it will be rerouted using a better route if available in **conf spvx spvcc pnni parameters reroute**. (See Section 12.1.2.4.5 in Part 2 of the *AMI Configuration Commands Reference Manual* for more information about this command.) **disable** indicates that this SPVC will not be retried, even if a DTL with a higher priority becomes available. The default state for PNNI SPVCs is **enable**.
-autoDtl (enable | disable)	**enable** means that dynamic path computation is used to route an SPVC when all of the configured DTLs fail. **disable** means that dynamic path computation is not used to route an SPVC when all of the configured DTLs fail. This feature applies to both FT-PNNI and PNNI DTLs. The default is **enable**. This flag does not apply when no DTLs are configured.
-backoff (enable | disable)	**enable** means the back-off mechanism is used for this point-to-point SPVC. Upon a failure, the failed directed DTL is not retried until the amount of time specified under **conf spvx spvcc pnni parameters backoff interval** has elapsed. This allows old information to age out and the network to stabilize. **disable** means the back-off mechanism is not used for this SPVC and directed DTLs that are configured for this SPVC are retried continually even after a failure. The default is **enable**. See Section 12.1.2.4 in Part 2 of the *AMI Configuration Commands Reference Manual* for more information.

Parameter	Description
-dtlTag <tag>	The Designated Transit List (DTL) tag, or index number, assigned to this SPVC which corresponds to a list of preferred DTLs that you configure under **`conf atmr dtltable`**. The DTL specifies the preferred call routing for the SVC portion of the SPVC. See Section 12.1.2.4 in Part 2 of the *AMI Configuration Commands Reference Manual* for more information.
-bearerClass (X | A | C)	The requested broadband bearer class for this point-to-point SPVC. `X` is for all types of ATM media. `A` is for non-ATM CBR media. `C` is for non-ATM VBR, UBR, and ABR media. The default is `X`.
-clip <no | yes>	During speech transmission, clipping is the loss of a brief interval at the beginning of a speech spurt. `no` indicates this SPVC is not susceptible to clipping. `yes` indicates this SPVC is susceptible to clipping. The default is `no`.
-QoSExpIndex <index>	The index number in the QoS expansion table that is used to do QoS class-to-parameter expansion when sending the SPVC call set-up message across a PNNI link. See the `QoSExpIndex` field under **`conf qos show`** for this number. The qosindex that is specified for this SPVC must already be present in the switch. The default is `0`.
-fqos (class0 | class1 | class2 | class3 | class4)	The requested QoS class for this point-to-point SPVC in the forward (calling to called) direction. The `-fqos` and `-bqos` options must be either both class0 or both a non-zero class. The default is `class0`.
-bqos (class0 | class1 | class2 | class3 | class4)	The requested QoS class for this point-to-point SPVC in the backward (called to calling) direction. The `-fqos` and `-bqos` options must be either both class0 or both a non-zero class. The default is `class0`.

[1] The forward and backward UPC contracts that you specify must be compatible with the `bearerClass` parameter for the SPVC (e.g., all parameters are for VBR traffic or CBR traffic).

NOTE

When creating a point-to-point SPVC, the `-fqos` and `-bqos` options must be either both `class0` or both a non-zero class. For example, the `-fqos` cannot be `class0` and the `-bqos` be `class1`. This combination is invalid. Having an `-fqos` of `class0` and a `-bqos` of `class0` is a valid combination, and having an `-fqos` of `class1` and a `-bqos` of `class4` is a valid combination, but having an `-fqos` of `class2` and a `-bqos` of `class0` is not valid.

NOTE

The *<destprefix*:*destport>* option has been removed. If the destination switch is running *ForeThought* 5.3.x or greater, you should use the *<destnsap>* option of specifying the destination. You can use the **conf spvx spvcc pnni destnsap** command on the destination switch to obtain the NSAP address to be used with the **conf spvx spvcc pnni pp new** command at the source switch. However, if the destination switch is running a pre-*ForeThought* 5.3.x release, you have to compute the destination NSAP address just as you did in previous releases.

The following is an example of how to create a simple PNNI point-to-point SPVC using the *<destnsap>* option. This method can only be used if the destination switch is running *ForeThought* 5.3.x or greater.

1. Open a session to the destination switch as follows:

```
myswitch::> open 198.29.22.46 private
Opening a session for “198.29.22.46”, please wait...
Connected to “198.29.22.46” (asx200bx).
*fishtank::>
```

2. Run the following command on the destination switch to obtain that switch’s NSAP address:

```
*fishtank::>conf spvx spvcc pnni destnsap 1 1d1
Destination Nsap : 0x47.000580ffe1000000f21a2ec0.0020480d0211.00
```

3. Go back to the source switch and create a new PNNI point-to-point SPVC. Copy the NSAP address into the command for the *<destnsap>* parameter:

```
*fishtank::> localhost
myswitch::>conf spvx spvcc pnni pp new 1a1 0 100
0x47.000580ffe1000000f21a2ec0.0020480d0211.00
```

1.5.4.3 Creating a PMP PNNI SPVC

To create a point-to-multipoint (PMP) PNNI SPVC, log in to AMI and enter the following parameters:

NOTE

You do not need to open a session to the destination switch to create a PNNI PMP SPVC because all PNNI SPVCs are bidirectional.

```
      conf spvx spvcc pnni pmp new <port> <vpi> <vci>
                       [-priority <priority>] \
                        [-pmpspvcid <index>] \
                           [-fupc <index>]
                           [-name <name>]
                         [-domainid <id>] \
    {-partynsap <nsap>  [-partyvpi <vpi> -partyvci <vci>]
            [-partyid <pid>]  [-partyname <name>]
                       [-ftpnniDtl  <dtlindex>]
                        [-QoSExpIndex <index>]
                                  }*
```

The advanced options are as follows:

```
[-bearerClass (X | A | C)] [-clip <no|yes>] \
 [-fqos (class0|class1|class2|class3|class4)]
```

NOTE: * Indicates ONE or more instances of the options (within the braces).

These parameters are defined as follows:

Parameter	Description
port	The port number where the permanent portion of the PMP SPVC resides on the source (local) switch fabric.
vpi	The virtual path number for the permanent portion of the PMP SPVC on the source (local) switch fabric.
vci	The virtual channel number for the permanent portion of the PMP SPVC on the source (local) switch fabric.

Parameter	Description
-priority <priority>	The order in which and the frequency with which you want this PMP SPVC to be tried for a connection. Higher priority PMP SPVCs are tried more often than lower priority ones. All parties within a PMP SPVC have the same priority. The value of **lowestpriority** determines the number of times each PMP SPVC is tried before attempting a lower priority PMP SPVC. Valid values are from **1** to **lowestpriority**, with 1 being the highest priority (the first and the most often tried) and with **lowestpriority** being the value configured under **conf spvx spvcc pnni parameters priority lowestpriority**. The default value is half of the value of **lowestpriority**, rounded off to the lowest integer. The value for **lowestpriority** is defaulted to 10. PMP and PP SPVC priorities are shared meaning some PMP or PP SPVCs can be assigned a higher priority compared to other PMP or PP SPVCs. For PMP and PP SPVCs with equal priority, PP SPVCs have higher priority over PMP SPVCs.
-pmpspvcid <index>	The index number assigned to this PMP SPVC to identify it uniquely. For PNNI PMP SPVCs, the index number may be assigned either by AMI or by the user. Valid values are integers between 1 and 65535.
-fupc <index>[1]	The forward (going from the local switch fabric to the remote switch fabric) UPC contract index assigned to this SPVC. To find the index you want, use the **conf upc show** command. If no index is specified, the default index of 0 (UBR best effort) is used.
-name <name>	An optional name for the user to assign to this PNNI PMP SPVC to help uniquely identify it.
-domainid <id>	The PNNI domain ID number on the source (local) switch fabric. The default is 1, which is the ID of the default routing domain.
-partynsap <nsap>	The NSAP address of the various destinations.
-partyvpi <vpi>	The virtual path number of the various destinations.
-partyvci <vci>	The virtual channel number of the various destinations.
-partyid <pid>	The index assigned to this party. If an ID is not specified, one will be assigned upon creation of the party.
-partyname <name>	The name assigned to this party.
-ftpnniDtl <dtlindex>	The Designated Transit List (DTL) tag, or index number, assigned to this PMP SPVC. The DTL specifies the preferred call routing for the SVC portion of the SPVC. Only one DTL can be configured. See in Part 2 of the *AMI Configuration Commands Reference Manual* for more information.
-QoSExpIndex <index>	The index number in the QoS expansion table that is used to do QoS class-to-parameter expansion when sending the PMP SPVC call set-up message across a PNNI link. See the QoSExpIndex field under **conf qos show** for this number. The qosindex that is specified for this PMP SPVC must already be present in the switch. The default is 0.
-bearerClass (X | A | C)	The requested broadband bearer class for this PMP SPVC. X is for all types of ATM media. A is for non-ATM CBR media. C is for non-ATM VBR, UBR, and ABR media. The default is X.
-clip <no | yes>	During speech transmission, clipping is the loss of a brief interval at the beginning of a speech spurt. no indicates this PMP SPVC is not susceptible to clipping. yes indicates this PMP SPVC is susceptible to clipping. The default is no.

Parameter	Description
-fqos (class0 | class1 | class2 | class3 | class4)	The requested QoS class for this PMP SPVC in the forward (calling to called) direction. The `-fqos` and `-bqos` options must be either both class0 or both a non-zero class. The default is `class0`.

[1] The forward and backward UPC contracts that you specify must be compatible with the `bearerClass` parameter for the PMP SPVC (e.g., all parameters are for VBR traffic or CBR traffic).

NOTE

When creating a PMP SPVC, the `-fqos` and `-bqos` options must be either both `class0` or both a non-zero class. For example, the `-fqos` cannot be `class0` and the `-bqos` be `class1`. This combination is invalid. Having an `-fqos` of `class0` and a `-bqos` of `class0` is a valid combination, and having an `-fqos` of `class1` and a `-bqos` of `class4` is a valid combination, but having an `-fqos` of `class2` and a `-bqos` of `class0` is not valid.

NOTE

The *`<destprefix`*`:`*`destport>`* option has been removed. If the destination switch is running *ForeThought* 5.3.x or greater, you should use the *`<destnsap>`* option of specifying the destination. You can use the **`conf spvx spvcc pnni destnsap`** command on the destination switch to obtain the NSAP address to be used with the **`conf spvx spvcc pnni pmp new`** command at the source switch. However, if the destination switch is running a pre-*ForeThought* 5.3.x release, you have to compute the destination NSAP address just as you did in previous releases.

The following is an example of how to create a simple PNNI PMP SPVC. This method can only be used if the destination switch is running *ForeThought* 5.3.x or greater.

1. Open a session to the destination switch as follows:

```
myswitch::> open 198.29.22.46 private
Opening a session for “198.29.22.46”, please wait...
Connected to “198.29.22.46” (asx200bx).
*fishtank::>
```

2. Run the following command on the destination switch to obtain that switch’s NSAP address:

```
*fishtank::> conf spvx spvcc pnni destnsap 1 1d1
Destination Nsap : 0x47.000580ffe1000000f21a2ec0.0020480d0211.00
```

3. Go back to the source switch and create a new PNNI PMP SPVC. Copy the NSAP address into the command for the **-partynsap** *<nsap>* parameter:

```
*fishtank::> localhost
myswitch::> conf spvx spvcc pnni pmp new 1a1 0 100
-partynsap 0x47.000580ffe1000000f21a2ec0.0020480d0211.00
```

For information about adding parties to a PMP SPVC, see Chapter 12 in Part 2 of the *AMI Configuration Commands Reference Manual.*

1.6 Traffic Types

NOTE

The following sections contain informal definitions of the concepts presented. For a detailed understanding of these issues, please see the ATM Forum’s UNI 3.1 and TM 4.0 Specifications.

Quality of Service (QOS) Management is based on the bandwidth parameters associated with a virtual connection and the class of service and ATM Adaptation Layer (AAL) used for that connection. In order to support voice, video, and data, the ATM Forum has defined four classes of service, or traffic types: Constant Bit Rate (CBR), Variable Bit Rate (VBR), Available Bit Rate (ABR), and Unspecified Bit Rate (UBR).

- At connection set-up time, traffic that uses a CBR parameter, such as a voice signal, makes a request for a dedicated Peak Cell Rate (PCR). Once the PCR is defined, the ATM network must be able to guarantee that amount of bandwidth for the duration of the connection.
- At connection set-up time, traffic that uses a VBR parameter, such as a video and data, makes a request for a dedicated PCR, Sustainable Cell Rate (SCR), and Maximum Burst Size (MBS). Once these cell rates are defined, the ATM network must be able to guarantee these rates for the duration of the connection.
- At connection set-up time, ABR traffic makes a request for a dedicated PCR and Minimum Cell Rate (MCR). Once these cell rates are defined, the ATM network must be able to guarantee the MCR rate for the duration of the connection. ABR traffic sources adjust their transmission rate in response to information they receive describing the status of the network and its capability to successfully deliver data.
- UBR traffic, such as broadcast information and ARP messages, is also known as “best effort” service. UBR provides no bandwidth guarantees.

Because ATM is designed to provide a single network to transport this variety of traffic classes, FORE’s traffic policing and Connection Admission Control (CAC) schemes are vital to allowing this mix of traffic to flow smoothly.

1.7 Traffic Policing (Usage Parameter Control)

Traffic policing, also known as Usage Parameter Control (UPC), is a method of ensuring fair allocation of network resources and of assessing the cells entering the switch for conformance with pre-established traffic bandwidth contracts. Those cells that exceed the specified contract are “tagged” or “dropped,” depending on what is defined in the contract. This ensures that the connections with reserved bandwidth are not exceeding their reservations. FORE Systems’ switches use a combination of “leaky bucket,” or Generic Cell Rate Algorithm (GCRA) hardware in the switch fabric and user-configurable parameters in AMI to perform these policing functions.

1.7.1 Leaky Bucket Algorithm

The first important concept to understand is the leaky bucket algorithm. Leaky buckets are a mechanism by which cells entering the switch fabric are monitored for compliance with UPC traffic contracts that have been negotiated at connection set-up time. Before the leaky buckets are discussed, it is important to understand the parameters that are being measured by the buckets, as shown in Table 1.1. These parameters are informally defined as follows:

- Peak Cell Rate (PCR) - the maximum number of cells per second
- Cell Delay Variation Tolerance (CDVT) - the tolerance for variation in the interarrival time of these cells, or the amount of jitter that can be accepted by the network
- Sustainable Cell Rate (SCR) - the average rate of cell transmission for this connection, taking bursting into account
- Burst Tolerance (BT) - the maximum amount of cells that can be transmitted
- Minimum Cell Rate (MCR) - the minimum rate that the network has to guarantee for an ABR connection

The leaky bucket algorithm is basically a timer which assesses if cells entering the switch fabric conform to the parameters listed above. As a cell arrives, the timer assesses if the cell is on time, late, or early. If the cell is determined to be on time or late (based on the traffic parameters), the cell is allowed to pass unchanged. If the cell is early (which, in turn, causes the cell stream to exceed the specified parameters), the cell is considered non-conforming and is either dropped or tagged (the CLP bit is set to 1), depending on the specified contract.

The first bucket in this analogy measures the PCR, or the rate at which the bucket drains. It also considers the CDVT, or the depth of the bucket. The second bucket measures the SCR, or the rate at which the bucket drains, and the BT, or the depth of the second bucket.

1.7.2 Non-conforming Cells: Tagging vs. Dropping

Second, it is important to understand the concept of tagging and dropping. Each ATM cell has a Cell Loss Priority (CLP) bit which indicates if the network can drop it under congested conditions. When the CLP bit is set to 0 (or CLP=0), the cell is assessed for compliance with traffic parameters associated with the CLP=0 stream. If the traffic parameters dictate that non-compliant cells should be “tagged,” the CLP bit is set to 1 (or CLP=1) by the UPC contract, which means that upon experiencing congestion further in the network, these CLP=1 cells are dropped in preference to CLP=0 cells.

Table 1.1 shows the various traffic contracts and user-configurable actions to be executed upon the detection of non-conforming cells.

Table 1.1 - Summary of Traffic Contract Variables and Policing Actions

Traffic Type	PCR	SCR	MBS	Bucket 1	Bucket 2	Policing on Switch Fabric	Action on non-conforming cells	
							Bucket 1	Bucket 2
CBR	PCR01			PCR01, CDVT		Yes	Drop CLP=0+1	
CBR0	PCR0, PCR01			PCR01, CDVT	PCR0, CDVT	Yes	Drop CLP=0+1	If enabled, Tag CLP=0; else, Drop CLP=0
VBR	PCR01	SCR01	MBS01	PCR01, CDVT	SCR01, BT01+CDVT	Yes	Drop CLP=0+1	Drop CLP=0+1
VBR0	PCR01	SCR0	MBS0	PCR01, CDVT	SCR0, BT0+CDVT	Yes	Drop CLP=0+1	If enabled, Tag CLP=0; else, Drop CLP=0

1.7.3 UPC Traffic Contract Parameters

The ATM Forum has defined different types of traffic contracts to be used in conjunction with these leaky buckets. The parameters that make up these types of contracts are defined as follows:

- pcr0 - PCR for cells with CLP=0
- pcr01 - PCR for the aggregate of the CLP=0 cells and the CLP=1 cells (all cells)
- scr0 - SCR for cells with CLP=0
- scr01 - SCR for the aggregate of the CLP=0 cells and the CLP=1 cells (all cells)
- mbs0 - MBS for cells with CLP=0
- mbs01 - MBS for the aggregate of the CLP=0 cells and the CLP=1 cells (all cells)
- tag - sets CLP bit=1 for CLP=0 cells that fail the PCR0 test for CBR0 contracts or the SCR0/MBS0 test for VBR0 contracts

The specific combinations of these parameters that make up the ATM Forum contracts are defined as follows:

1. cbr <pcr01>
2. cbr0 <pcr0> <pcr01> [tag]
3. vbr <pcr01> <scr01> <mbs01>
4. vbr0 <pcr01> <scr0> <mbs0> [tag]
5. abr <pcr01> <mcr>

The `cbr <pcr01>` contract is for CBR traffic. It only uses the first leaky bucket to assess the conformance to PCR of the aggregate of the CLP=0 cells and the CLP=1 cells. Cells which fail the PCR CLP=0+1 test are discarded.

The `cbr0 <pcr0> <pcr01> [tag]` contract is for CBR traffic. It uses the first leaky bucket to assess the conformance to PCR of the CLP=0 cells. It uses the second leaky bucket to assess the conformance to PCR of the aggregate of the CLP=0 and the CLP=1 cells. If the `tag` option is set, the cells which fail the PCR CLP=0 test are tagged as CLP=1 and passed on to the second leaky bucket to be tested for PCR CLP=0+1 conformance. Cells which fail the PCR test on CLP=0+1 are discarded. If the `tag` option is not set, cells which fail the PCR CLP=0 test are discarded and cells which fail the PCR test on CLP=0+1 are discarded.

The `vbr <pcr01> <scr01> <mbs01>` contract is for VBR traffic. The first leaky bucket assesses the conformance to PCR of the aggregate of CLP=0 cells and the CLP=1 cells and the second leaky bucket assesses the conformance to SCR and BT of this same combination. Cells which fail the PCR test are dropped. Cells which pass the PCR test, but which fail SCR and BT test are dropped.

The `vbr0 <pcr01> <scr0> <mbs0> [tag]` contract is for VBR traffic. It uses the first leaky bucket to assess the conformance to PCR of the aggregate of the CLP=0 and the CLP=1 cells. Cells that fail this test are discarded. It uses the second leaky bucket to assess the conformance to SCR and BT of the CLP=0 cells. If the `tag` option is set, the cells which fail the SCR and BT CLP=0 test are tagged as non-conforming cells. If the `tag` option is not set, cells which fail the SCR and BT CLP=0 test are discarded.

The `abr <pcr01> <mcr>` contract is for ABR traffic. It only uses the first leaky bucket to assess the conformance to PCR of the aggregate of the CLP=0 cells and the CLP=1 cells. Cells which fail the PCR CLP=0+1 test are discarded. The MCR is the guaranteed minimum rate and the PCR is the maximum required rate. The MCR may be set to 0, which indicates “best effort” service. If MCR is set greater than 0, then the rate is guaranteed, up to MCR. However, between MCR and PCR, cells may be dropped when congestion is experienced.

1.7.4 AMI UPC Commands

AMI allows you to create a UPC contract using these combinations of traffic parameters. To create a UPC contract in AMI, enter the following parameters:

```
configuration upc new <index> ubr [aal5 [noPktDisc]] [ubrTagging]
        [AltCLP] [-estbw <kcps>] [-name <name>]
```

OR

```
configuration upc new <index> abr <pcr01> <mcr> [-cdvt <us>] [noGCRA]
        [aal5 [noPktDisc] [PPPol]] [-name <name>]
```

OR

```
configuration upc new <index> <UPC> [-cdvt <us>] [noGCRA]
        [aal5 [noPktDisc] [PPPol]] [AltCLP]
[-scheduling (roundrobin | smoothed | guaranteed)] [-name <name>]
```

```
Where UPC is one of the following combinations of traffic parameters:
```

```
cbr  <pcr01>
cbr0 <pcr0> <pcr01> [tag]
vbr  <pcr01> <scr01> <mbs01>
vbr0 <pcr01> <scr0> <mbs0> [tag]
```

```
Advanced options (to generate UPC contract from Frame Relay parameters):
```

```
[[-bc <bits>][-be <bits>][-cir <kbps>][-ar <kbps>][-frsize <bytes>]]
```

These parameters are defined as follows:

Parameter	Description
index	The integer index that refers to this specific traffic contract. Valid index numbers are from 0 to 32,767.
UPC	One of the types of traffic contracts shown above. The parameters in these contracts are defined as follows:
ubr	Indicates UBR traffic.
cbr[1]	Indicates CBR traffic.
cbr0	Indicates CBR0 traffic.
vbr	Indicates VBR traffic.
vbr0	Indicates VBR0 traffic.
pcr0	Indicates the peak cell rate for cells with CLP = 0.
pcr01	Indicates the peak cell rate for all cells.

Parameter	Description
scr0	Indicates the sustainable cell rate for cells with CLP = 0.
scr01	Indicates the sustainable cell rate for all cells.
mbs0	Indicates the maximum burst size for cells with CLP = 0.
mbs01	Indicates the maximum burst size for all cells.
tag	**tag** means that non-conforming CLP = 0 cells are tagged. Otherwise, they are dropped. The default is that they are dropped. This option only applies to the PCR0 parameter of the CBR0 contract and to the SCR0 and MBS0 parameters of the VBR0 contract.
abr	Indicates ABR traffic. Currently, ABR UPC contracts are supported only on Series D network modules.
mcr	Indicates the minimum cell rate for all cells. ABR connections with an MCR equal to 0 use the roundrobin scheduling discipline. ABR connections with an MCR greater than 0 use the guaranteed scheduling discipline. However, if there are no suitable rate groups in the rate controller, the ABR connections with an MCR greater than 0 are rejected. Currently, ABR UPC contracts are supported only on Series D network modules.
-cdvt us	The Cell Delay Variation Tolerance (CDVT) associated with the peak cell rates, in microseconds. If the CDVT is not specified here, the default CDVT value associated with the port will be used. (See **conf port show** and **conf port cdvt** for more information).
noGCRA	**noGCRA** means that GCRA policing is disabled on CBR or VBR (depending on what is configured) connections using this contract. If **noGCRA** is not entered, then GCRA policing is enabled on CBR or VBR (depending on what is configured) connections using this contract. By default, **noGCRA** is not entered (GCRA policing is enabled). You must use the **noGCRA** option when applying a UPC contract to the outbound signalling channel using the **-outsigupc** *<upc-index>* variable under **conf signalling new** to prevent the outbound signalling channel from being policed.
aal5	The connection is using the AAL5 Adaptation Layer.
noPktDisc	This optional parameter can only be used if the connection is AAL5 (i.e., the **aal5** parameter is present). This parameter suppresses EPD/PPD (AAL5 packet discard) on the connection. The default is for this parameter not to be present (EPD/PPD is enabled) on AAL5 connections.
ubrTagging	**ubrTagging** means that all UBR traffic is tagged (set to CLP=1) on this connection. If **ubrTagging** is not entered, then UBR traffic is not tagged on this connection. This command only applies to UBR traffic. By default, UBR traffic is not tagged.
PPPol[2]	This optional parameter can only be used if the connection is AAL5 (i.e., the **aal5** parameter is present). This parameter indicates that Partial Packet Policing is going to be performed on this connection. The default is for this parameter not to be present, which leaves Partial Packet Policing disabled.
AltCLP	This optional parameter only applies to connections on Series D network modules. It indicates that the alternate CLP threshold (configured using **conf module traffic d altclpthresh**) should be used for all connections created with this UPC contract. The default is for this parameter not to be present, which means the connections will not use the alternate CLP threshold.

Parameter	Description
-estbw <kcps>	Applies only to UBR SPVCs. Lets you assign an estimated bandwidth value, in kilocells per second, for individual UBR SPVCs. The value must be an integer from 0 to 65534. This value is used by the load-balanced UBR routing feature. See Part 1 of the *AMI Configuration Commands Reference Manual* for more information about load-balanced UBR routing.
-scheduling (roundrobin | smoothed | guaranteed)[3]	Indicates the scheduling mode to be used for servicing traffic on the output side of a Series D network module. **roundrobin** means that all service for these connections comes from one of the round-robin queues in the network module. This is the default mode for both SVCs and PVCs. **smoothed** means that all service for these connections comes from the network module's rate controller, which ensures that cells for these connections are transmitted into the network at a fixed rate of R cells per second. **guaranteed** is a combination of the round-robin and smoothed modes. Service for these connections are scheduled with both fixed rate R from the rate controller, and they have an entry in the appropriate round-robin queue.
-name <name>	The user-defined name associated with this UPC traffic contract. This helps you remember for what traffic type this specific contract is used. If you do not specify a name, a default name that relates to this type of traffic contract is assigned automatically.
-bc <bits>	The committed burst size of a connection, in bits. Can only be used on a Frame Relay connection.
-be <bits>	The excess burst size of a connection, in bits. Can only be used on a Frame Relay connection.
-cir <kbps>	The committed information rate of a connection, in kbps. Can only be used on a Frame Relay connection.
-ar <kbps>	The access rate of a Frame Relay UNI, in kbps. Can only be used on a Frame Relay connection.
-frsize <bytes>	The average frame size, in bytes. Can only be used on a Frame Relay connection.

[1] The units for **pcr0**, **pcr01**, **scr0**, **scr01**, **mbs0**, and **mbs01** are specified either in cells per second or in kilobits per second, depending on what you used for **conf system units**. To display the current setting, use **conf system show**. The default is **cps** (cells per second).

[2] The HDCOMP ASIC must be version 1 or greater to support AAL5 partial packet policing. To display the ASIC version, use the **display hdcomp** command.

[3] The **-scheduling** option has an effect only on connections with outputs on Series D network modules. All other network module platforms only use **roundrobin** scheduling.

NOTE

Remember, when you create a UPC contract, it is not actually used until you assign it to a VPC, VCC, or SPANS path. UPC contracts are not assigned to VPTs.

NOTE

When the advanced options are used, they are converted to ATM UPC parameters and are displayed as ATM parameters.

The following is an example of how to create a UPC contract:

```
myswitch::configuration upc> new 5 vbr0 500 200 250 -cdvt 1000 aal5 PPPol -name vbr0_upc
```

This example specifies a contract named “vbr0_upc”, which is a VBR0 contract with an index of 5, a pcr01 of 500 cells / sec (or kbps), an scr0 of 200 cells / sec (or kbps), an mbs0 of 250 cells (or kilobits), a CDVT of 1,000 microseconds, and partial packet policing enabled.

NOTE

For more information regarding traffic contracts, please refer to Table 5-7 in the ATM Forum UNI 3.0 Specification.

NOTE

PVCs that use UPC contracts containing any of the `[noGCRA]`, `[aal5 [noPktDisc] [PPPol]]`, and `[ubrTagging]` options are valid only when the **`conf port gcrapolicing`**, **`conf port aal5packetdiscard`**, **`conf port pppolicing`**, and **`conf port ubrtagging`** commands are set to `svcOn` or `svcOff`. Use **`conf port show tm`** to check these settings.

CHAPTER 2 Configuring Classical IP

2.1 Introduction

This chapter describes how to design, configure, and maintain a Classical IP ATM network. The term classical indicates that the ATM network has the same properties as existing legacy LANs. That is, even though ATM technology allows for large, globally connected networks, for example, it is only used in the LAN environment as a direct replacement of existing LAN technology. The classical model of LANs connected through IP routers is maintained in ATM networks. RFC-1577 provides the standard for Classical IP over ATM.

Classical IP over ATM is different than IP in legacy LANs in that ATM provides a virtual connection environment through the use of Permanent Virtual Circuits (PVCs) and/or Switched Virtual Circuits (SVCs). SVC management is performed via the ATM Forum UNI 3.0 Specification, which specifies Q.2931. Q.2931 is a broadband signalling protocol designed to establish connections dynamically at the User-Network Interface (UNI). Q.2931 uses Service Specific Connection Oriented Protocol (SSCOP) as a reliable transport protocol, and all signalling occurs over VPI: 0, VCI: 5. Q.2931 connections are bidirectional, with the same VPI/VCI pair used to transmit and receive.

Once a Classical IP connection has been established, IP datagrams are encapsulated using IEEE 802.2 LLC/SNAP and are segmented into ATM cells using ATM Adaptation Layer type 5 (AAL5). Additionally, the default Maximum Transmission Unit (MTU) is 9,180 bytes (the SNAP header adds 8 more bytes) with a maximum packet size of 65,535 bytes. There is currently no support for IP broadcast datagrams or IP multicast datagrams in a Classical IP environment.

2.1.1 Logical IP Subnets

An important concept in Classical IP networks is that of a Logical IP Subnet (LIS). An LIS is a group of hosts configured as members of the same IP subnet (that is, they have the same IP network and subnetwork numbers). In this sense, one LIS can be equated to one legacy LAN. It is possible to maintain several overlaid LISs on the same physical ATM network. Therefore, in a Classical IP ATM network, placing a host on a specific subnet is a logical choice rather than a physical one. In this type of environment, communication between hosts in different LISs is only permitted by communicating through an IP router which is a member of both LISs (as per RFC-1577).

The number of LISs, and the division of hosts into each LIS, is purely an administrative issue. Limitations of IP addressing, IP packet filtering, and administrative boundaries may guide a manager into establishing several LISs onto a single ATM network. Keep in mind, though, that communication between LISs must occur through IP routers.

2.1.2 Classical IP Interfaces

In order to support routing between multiple LISs, the switch software allows a switch to be configured as a member of (and a router between) up to four distinct LISs. (The host adapter software allows a host to be configured as a member of (and a router between) up to 16 distinct LISs.) Each LIS membership is through a separate Classical IP network interface. Existing system level IP routing configuration tools are used to control routing through each of the Classical IP interfaces in the same manner as routing among several physical interfaces. Even though each Classical IP interface associated with a given physical interface uses the same physical hardware, they are each configured separately with their own MTU, IP address, and ATM address.

By default, the name of each of the Classical IP interfaces on a switch begins with **qa**. (On a host, each of the Classical IP interfaces begins with **ci** and is user-configurable.) On a switch, all of the Classical IP interfaces associated with physical unit zero have **a** as the next letter. All of the Classical IP interfaces associated with physical unit one have **b** as the next letter, and so forth. Finally, each Classical IP interface has its interface number as a suffix. As an example of the above naming convention for switches, the name of the third Classical IP interface (unit 2) on physical unit one is **qab2**.

2.1.3 SPANS Interface

While each of the Classical IP interfaces for a given physical interface is designed to support Classical IP using Q.2931 signalling, a SPANS interface also exists for each physical interface. The SPANS interface is **`asx0`** in switch software and is **`fa0`** in host adapter software.

The SPANS interface supports FORE IP on top of SPANS signalling. FORE IP allows communication using AAL4 or AAL5 with no encapsulation, uses a broadcast ARP for SPANS address resolution, and supports direct communication of all hosts on a physical ATM network without the use of IP routers. Since SPANS and Q.2931 signalling use different VCIs, a host can simultaneously support FORE IP over SPANS as well as Classical IP over Q.2931 on the same physical interface.

As a result of standard IP routing, all traffic sent out a SPANS interface uses FORE IP, while all traffic sent out a Classical IP interface uses Classical IP. The Classical IP interfaces are **`qaaX`** (where X is 0 - 3) in switch software and are **`ci`** in host adapter software.

Each of the SPANS interfaces should be assigned an IP address on a subnet different than the subnets of any of the Classical IP interfaces. It is permissible to place multiple SPANS interfaces on the same subnet, and the driver load balances connections across these interfaces.

It is only necessary to configure the SPANS and Classical IP interfaces if the specific service provided by that interface is required. A host sending only Classical IP would not need to configure the SPANS interfaces. Likewise, a host sending only FORE IP would not need to configure the Classical IP interfaces. Both the SPANS and Classical IP interfaces may be configured simultaneously, but they must be in separate subnets. Remember that Classical IP specific configuration changes can only be done with the Classical IP devices, while SPANS specific configuration changes can only be done with the SPANS devices.

2.2 Address Registration and ILMI

Before a host can establish connections over a physical interface, the host must know the NSAP address for that interface. The primary purpose of Interim Local Management Interface (ILMI) is to discover and register these NSAP addresses dynamically.

2.2.1 NSAP Addresses

For private ATM networks, addresses uniquely identify ATM endpoints. The UNI 3.0 address format is modeled after that of an OSI Network Service Access Point, hence the name NSAP address.

Three address formats have been specified: DCC, ICD, and E.164. FORE implements the ICD ATM format. Per the UNI 3.0 specification, all private networks should accept initial call set-up messages containing ATM addresses with any of the approved formats and forward the calls as necessary.

An NSAP address consists of the following:

- a 13-byte network-side prefix - The prefix is the NSAP prefix of the switch to which the host is attached.
- a seven-byte user-side part - This consists of the following:
  - a six-byte End System Identifier (ESI) - The ESI is the unique IEEE MAC address of the interface.
  - a one-byte selector - Although each Classical IP interface for a given physical interface uses the same prefix and ESI, the selector field is the part that indicates the number of the specific Classical IP interface. On a switch, the selector field is 00 for qaa0, 01 for qaa1, 02 for qaa2, and 03 for qaa3.

2.2.2 Operating with ILMI Support

FORE Systems switches running *ForeThought* software provide support for ILMI. If ILMI is supported on all of the switches and hosts in a given network, when a switch boots up, ILMI enables the switch to discover all of the hosts attached to it and to send its ATM prefix associated with the port to those hosts dynamically. In return, the host prepends that prefix to its ESI and selector fields, forming a complete ATM address. The host then notifies the switch of its complete ATM address. These registration messages are sent and received over AAL5 using VPI: 0, VCI: 16. Once ILMI registration has been completed, then connection setup can occur.

If a host changes network ports after an ATM address has been registered for its interface, all existing connections are closed. If the new port is on a different switch, a new ATM address (with a different network address prefix) is registered. The host can then begin to establish new connections.

2.2.3 Operating without ILMI Support

If ILMI is not supported on a particular switch or host in a given network, then the ATM addresses must be manually configured. If another vendor's switch does not support ILMI, it can not supply an ATM prefix to the hosts. Therefore, the user must assign a unique, valid prefix to the switch. Additionally, the same prefix should be used for all hosts attached to that switch.

On the host, **`uniconfig`** is used to configure the ATM address for a specific interface. The switch directly attached to this interface is then informed of this ATM address/port combination through commands in the ATM Management Interface (AMI). Once the host and network have both been informed of this ATM address/port pair, the host may begin signalling.

2.2.4 Configuration

The choice to use ILMI for address registration is made at software installation time. Since ILMI uses SNMP as its management protocol, the use of ILMI is tied into **snmpd**. The choice can be made to run FORE's SNMP agent and use ILMI (**snmpd**), run FORE's SNMP agent without using ILMI (**snmpd `-n`**), or just use ILMI (**snmpd `-i`** or **`ilmid -i`**).

2.3 ARP and ARP Servers

2.3.1 Theory

In order for a host to establish a connection to another host, it must first determine the other host's ATM address. ATM ARP (ATM address resolution protocol) is the procedure used to resolve an IP address into an ATM address. Since the ATM standards do not currently support broadcast on an ATM LAN, address resolution is performed by direct communication with a special ARP server, rather than broadcasting ARP requests as is done in legacy LANs. Each LIS must have only one ARP server configured, but a single ARP server can be the server for several LISs.

Each host in an LIS must be configured with the ATM address of the host providing ARP service for its LIS. On a switch, the ATM address of the ARP server can be obtained by using the AMI command **`conf atmarp arpserver show`**. On a host, the ATM address of the ARP server can be obtained by using **`clipconfig show`** (remember to use the interface associated with the given LIS).

On a switch, the ATM address of the ARP server can be configured by using the instructions found in Section 2.3.2 of this manual. On a host, the ARP server address is configured into the host at installation time using the configure_atm script.

Since only one ARP server can be functioning at a time in a given LIS, and since the ARP server's address is configured into each host, it is not possible to use multiple, redundant ARP servers to improve robustness. If an ARP server becomes nonfunctional, a new ARP server must be configured, and then each host within the LIS must be configured to use the new ARP server. To configure a new ARP server address on a switch, use the the instructions found in Section 2.3.2 of this manual. To configure a new ARP server address on a host, you must delete the interface and recreate it.

2.3.2 Configuring a FORE Switch to be an ARP Server

FORE's ATM switches also have the capability of being an ARP server. (FORE's ATM adapters can not be configured as an ARP server.) To configure a FORE ATM switch as an ARP server, perform the following steps on only one of the SCPs:

1. On one of the SCPs, determine the ATM address of that SCP for the relevant interface (qaa0 -> qaa3) using the following AMI command:

```
configuration atmarp getnsap <interface>
```

For example:

```
configuration atmarp getnsap qaa0

qaa0 NSAP address: 47000580ffe1000000f12400de0020481900de00
```

2. Set the NSAP address of the ARP server to be the ATM address of the interface that you displayed in step 1 using the following AMI command:

```
conf atmarp arpserver set <NSAPaddress> [<interface>]
```

For example:

```
conf atmarp arpserver set 0x47000580ffe1000000f12400de0020481900de00 qaa0
```

3. Set the ATM address of the ARP server on each of the other switches that will use that switch as the ARP server using the same command found in step 2.

2.3.3 Classical IP Operation

Once a host knows its own ATM address and the ATM address of its ARP server it attempts to establish a connection to the ARP server, which is used to send ARP requests and receive ARP replies. When the connection to the ARP server has been established, the ARP server sends an inverse ARP (InARP) request on the new VC to learn the host’s IP address. When an InARP reply is received, the ARP server places that host’s IP address to ATM address mapping in its ARP cache. Therefore, over time, the ARP server dynamically learns the IP-to-ATM address mappings of all the hosts in its LIS. It can then respond to ARP requests directed toward it for hosts in its LIS.

NOTE

In order for a host to communicate with an ARP server, it must have learned its own ATM address and have been configured with the ATM address of the ARP server.

A host can not resolve the ATM addresses of other hosts in its LIS unless it can communicate with its ARP server.

Since there is no mechanism for ARP servers to exchange mapping information with each other, it is imperative that each LIS be configured with only one ARP server.

When a host wants to communicate with another host in its LIS, it first sends an ARP request to the ARP server containing the IP address to be resolved. When an ARP reply is received from the ARP server, the host creates an entry in its ARP cache for the given IP address and stores the IP-to-ATM address mapping. This ARP cache entry is marked as complete. To ensure that all of the IP-to-ATM address mappings known by a certain host are up-to-date, hosts are required to age their ARP entries. A host must validate its ARP entries every 15 minutes (20 minutes on an ARP server). Any ARP entries not associated with open connections are immediately removed.

A host validates its SVCs by sending an ARP request to the ARP server. A host validates its PVCs, and an ARP server validates its SVCs, by sending an InARP request on the VC. If a reply is not received, the ARP entry is marked invalid. Once an ARP entry is marked invalid, an attempt is made to revalidate it before transmitting. Transmission proceeds only when validation is successful. If a VC associated with an invalid ARP entry is closed, the entry is removed.

2.3.4 Operational Issues

Certain hosts in an LIS may not support Classical IP. It is still possible to communicate with these hosts (and for these hosts to communicate with one another) by using static ARP entries. If a host does not support Classical IP, its IP-to-ATM address mapping should be placed in its ARP server's cache as a static entry. This allows other hosts that do support Classical IP to contact their ARP server as usual and obtain the correct address mapping. If a host that does not support Classical IP wants to initiate connections, the IP-to-ATM address mappings of the destination hosts should be put in its ARP cache, again as static entries. By using static ARP entries in the above fashion, the ability for all hosts to communicate is maintained.

There are some restrictions on the number of hosts that can be maintained dynamically. They are as follows:

- In the default configuration, a host can only have approximately 250 virtual connections open simultaneously. This means that an ARP server can only serve 250 clients, since each client must maintain a connection with its ARP server. This may be a limitation if the ARP server is servicing multiple LISs.
- It is possible to increase the number of connections using AMI.
- Some hosts may be limited to supporting a maximum of 1,024 connections per adapter.

2.4 Classical IP PVCs

2.4.1 Theory and Configuration

Normally, ATM connections in a Classical IP environment are established dynamically using UNI 3.0 or UNI 3.1. ARP, ILMI, and UNI 3.0 or UNI 3.1 all work together as described previously to set up an SVC. If a host from another vendor does not support Classical ARP or ILMI, it is still possible to set up an SVC using work-arounds. If a host or a switch in an LIS does not support UNI 3.0 or UNI 3.1, however, it is not possible to establish an SVC. In this case, a Classical IP PVC can be used for communication.

On each of the hosts, **`cliparp add -pvc`** is used to establish the PVC. An unused VPI/VCI pair must be chosen for each host. PVCs using the chosen VPI/VCI pairs must also be set up from each of the hosts to their connecting switch, and then on all of the switches between the two connecting switches.

NOTE

Both the incoming and outgoing connections are set up simultaneously on the host, but they must be set up individually on the switches. The same VPI/VCI pair is used by a host to send on the PVC as well as receive on the PVC. The IP datagrams are sent over the PVC using AAL5 with LLC/SNAP encapsulation.

2.4.2 Revalidation and Removal

Normally, the device driver periodically checks that its PVCs are still established and functioning. A host revalidates a PVC by sending InARP requests over the PVC, if the user specifies that revalidation should occur by using the **`-reval`** option to **`cliparp add -pvc`** at the time the PVC is created (e.g., **`-reval`** 10 means revalidation will occur every 10 minutes). If the equipment attached to the FORE equipment supports revalidation, the user must choose the **`-reval`** option. If an InARP reply is not received, the revalidation fails, the PVC is marked invalid (as shown through **`cliparp show`**), and communication over the PVC is no longer possible.

Once a PVC is marked invalid, an attempt is made to validate the PVC before transmitting. Transmission proceeds only when validation is successful. It is possible to disable this revalidation feature by not specifying the **`-reval`** option. This is often desirable when the remote end of the PVC (such as a video camera) does not support InARP.

A Classical IP PVC is removed on the host side using **`cliparp delete -pvc`**. Both the incoming and outgoing connections are removed simultaneously. The PVC must then be removed from each of the network switches involved.

2.5 Configuring the Network

In an ATM network, before any connections can be made, the two parties must know each other's ATM address in order to set up that connection.

To allow those connections to work, the ideal scenario is for all hosts and switches in the network to have support for both ILMI and for RFC-1577 (Classical IP over ATM). However, when using other vendors' equipment, this may not be the case. This section describes how to configure a network with the following scenarios:

- Configuring a third-party host that has no ILMI and no RFC-1577 support
- Configuring a third-party switch that has ILMI support, but no RFC-1577 support
- Configuring a third-party switch that has no ILMI support, but has RFC-1577 support

2.5.1 Third-Party Host with No ILMI and No RFC-1577 Support

To configure a network with a third-party vendor's host (or an edge device) that supports neither ILMI nor RFC-1577 (as shown in Figure 2.1), perform the following steps:

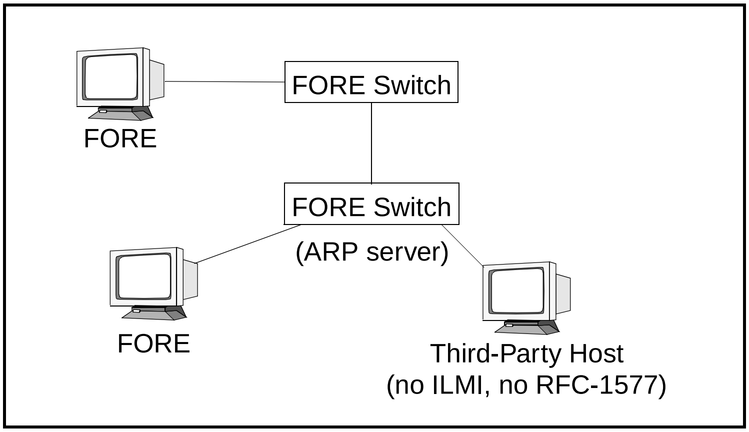

Figure 2.1 - Configuring a Third-Party Host with No ILMI and No RFC-1577 Support

1. Before beginning this process, be sure that *ForeThought* software is installed and running on the FORE equipment.
2. Using the configuration software of the third-party host, assign that host an NSAP address that has the same prefix as the switch fabric to which it is connected.
3. Configure the switch that is the ARP server so that it has a static route to the third-party host using the following AMI command:

```
conf atmr ftpnni staticroute new <NSAP> <mask> -port <port> -vpi <vpi>
```

Be sure to use a host mask value of 152.

2.5.2 Third-Party Switch with ILMI and No RFC-1577 Support

To configure a network with a third-party vendor's switch that supports ILMI, but not RFC-1577, (as shown in Figure 2.2), perform the following steps:

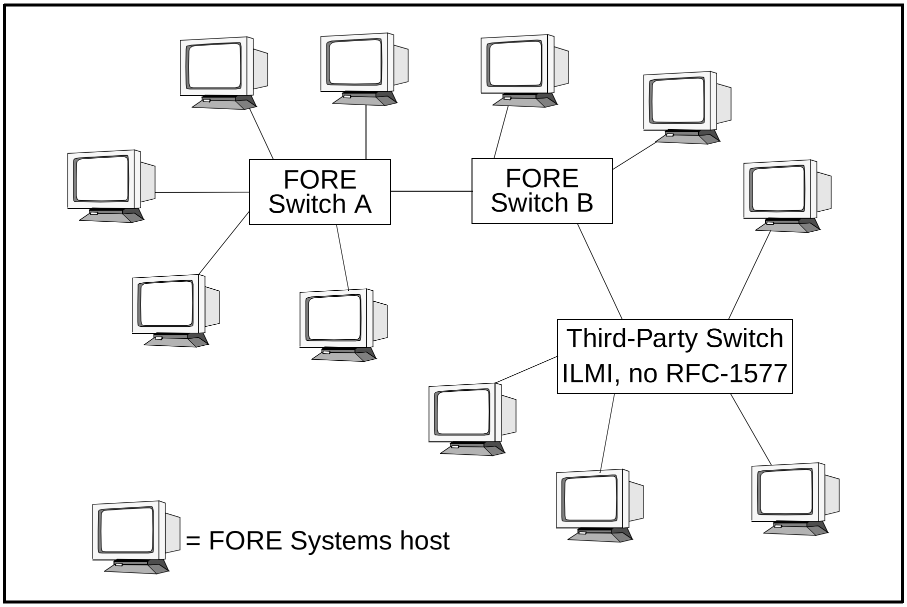

Figure 2.2 - Configuring a Third-Party Switch with ILMI Support and No RFC-1577

1. Be sure that *ForeThought* software has been installed on all of the hosts and that ILMI was set in the process. ILMI dynamically performs address registration for all of the hosts.
2. Configure a static ATM route to the third-party switch on FORE switch "B" that is physically connected to the third-party switch using the following AMI command:

```
conf atmr ftpnni staticroute new <NSAP> <mask> -port <port> -vpi <vpi>
```

   Be sure to use a network mask value of 104.

3. Configure two static NSAP routes on the third-party switch, one to each of the FORE switches to which the third-party switch is connected, using the third-party vendor's configuration software. Be sure to use a network mask value of 104.

2.5.3 Third-Party Switch with RFC-1577 and No ILMI Support

To configure a network with a third-party vendor's switch that does not support ILMI, but does support RFC-1577 (as shown in Figure 2.3), perform the following steps:

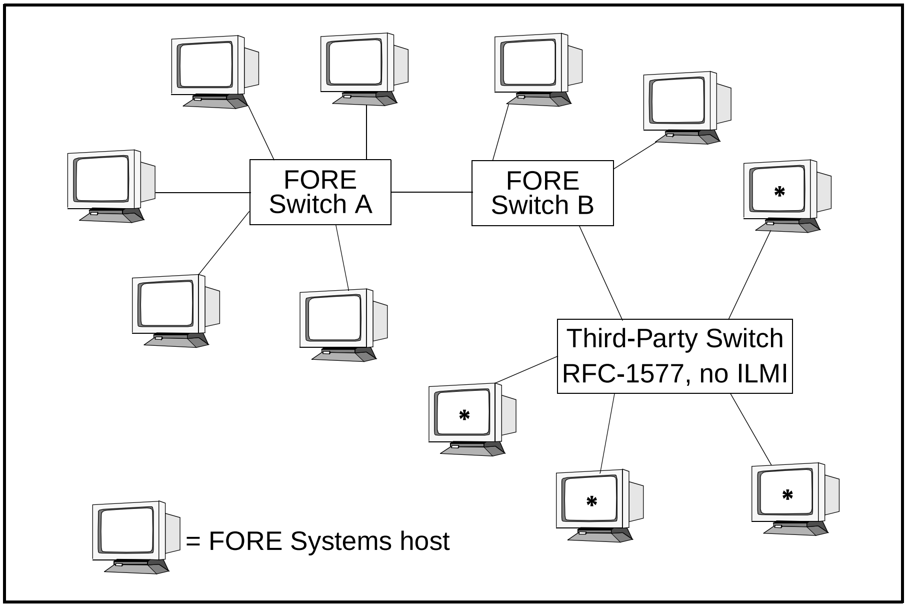

Figure 2.3 - Configuring a Third-Party Switch with RFC-1577 and No ILMI Support

1. Be sure that *ForeThought* software has been installed on all of the FORE hosts and that ILMI was set in the process. ILMI dynamically performs address registration for all of the FORE hosts and FORE switches.
2. Statically configure the non-FORE (*) hosts with ATM addresses (edit the firmware download script), using the same switch prefix for all of the hosts.
3. Configure a static ATM route to the third-party switch on FORE switch "B" that is physically connected to the third-party switch using the AMI command:

```
conf atmr ftpnni staticroute new <NSAP> <mask> -port <port> -vpi <vpi>
```

Be sure to use a network mask value of 104. Also, be sure to use the same prefix that was used to configure the hosts.

4. Configure two static ATM routes on the third-party switch, one to each of the FORE switches to which the third-party switch is connected, using the third-party vendor's configuration software. Be sure to use a network mask value of 104.

CHAPTER 3 Configuring an Emulated LAN

3.1 Introduction

This chapter describes how to design, configure, and maintain an Emulated LAN (ELAN) over an ATM network. An ELAN provides communication of user data frames among all members of the ELAN, similar to a physical LAN. One or more ELANs may run simultaneously (and independently) on the same ATM network.

Each ELAN is composed of a set of LAN Emulation Clients (LECs), a LAN Emulation Configuration Server (LECS), and at least one LAN Emulation Server (LES) and Broadcast and Unknown Server (BUS) pair (also referred to as colocated BUS or an intelligent BUS). In the current software release, the LECS may reside either in a *ForeRunner ASX*-200WG, *ASX*-200BX, *ASX*-1000, *ASX*-1200, *ASX*-4000, LE 25, LE 155, *ESX*-3000, *TNX*-210, or *TNX*-1100 switch, or in a UNIX workstation running Solaris 2.5, 2.5.1, or 2.6. The LES/BUS pair may reside either in a *PowerHub* 7000, *PowerHub* 8000, *ASN*-9000, *ASX*-200WG, *ASX*-200BX, *ASX*-1000, *ASX*-1200, *ASX*-4000, LE 25, LE 155, *ESX*-3000, *TNX*-210, or *TNX*-1100 switch, or in a UNIX workstation running Solaris 2.5, 2.5.1, or 2.6. An additional software feature is Distributed LAN Emulation (DLE), which provides load sharing and fault tolerance to the ELAN.

The current software release supports the emulation of both Ethernet (IEEE 802.3) and Token Ring ELANs.

3.1.1 Ethernet ELANs

The current software release supports emulation of Ethernet (IEEE 802.3) ELANs. In the current release, each LEC resides on an ATM host system (PC, Macintosh, UNIX workstation, ATM switch, *PowerHub* 7000, or ES-3810).

3.1.2 Token Ring ELANs

The current software release supports emulation of Token Ring (IEEE 802.5) ELAN services. In the current release, Token Ring LECs can not be created on the switch. Token Ring LECs can only be created on some hosts.

3.2 ELAN Components

The components of an ELAN include LECs, and LAN Emulation services consisting of a LECS, a LES, and a BUS. Although the ATM Forum specification allows the LES and BUS to be located on different devices, more intelligent traffic handling is possible when they are located on the same device. *ForeThought* 5.0.x or greater software requires the LES and BUS be co-located (reside on the same device).

The LECS may reside in the same physical system as the LES/BUS or in a separate physical system. For example, the LECS could reside in a switch, while the LES/BUS reside in a workstation. Currently, the LECS is supported only on *ForeRunner and TNX* switches and on systems running Solaris. The LES/BUS are supported only on *ForeRunner and TNX* switches, on *PowerHub* 7000s, on *PowerHub* 8000s, on ASN-9000s, and on systems running Solaris. The functional interconnections of a simple ELAN consisting of two LECs, an LECS, a LES, and a BUS are shown in Figure 3.1.

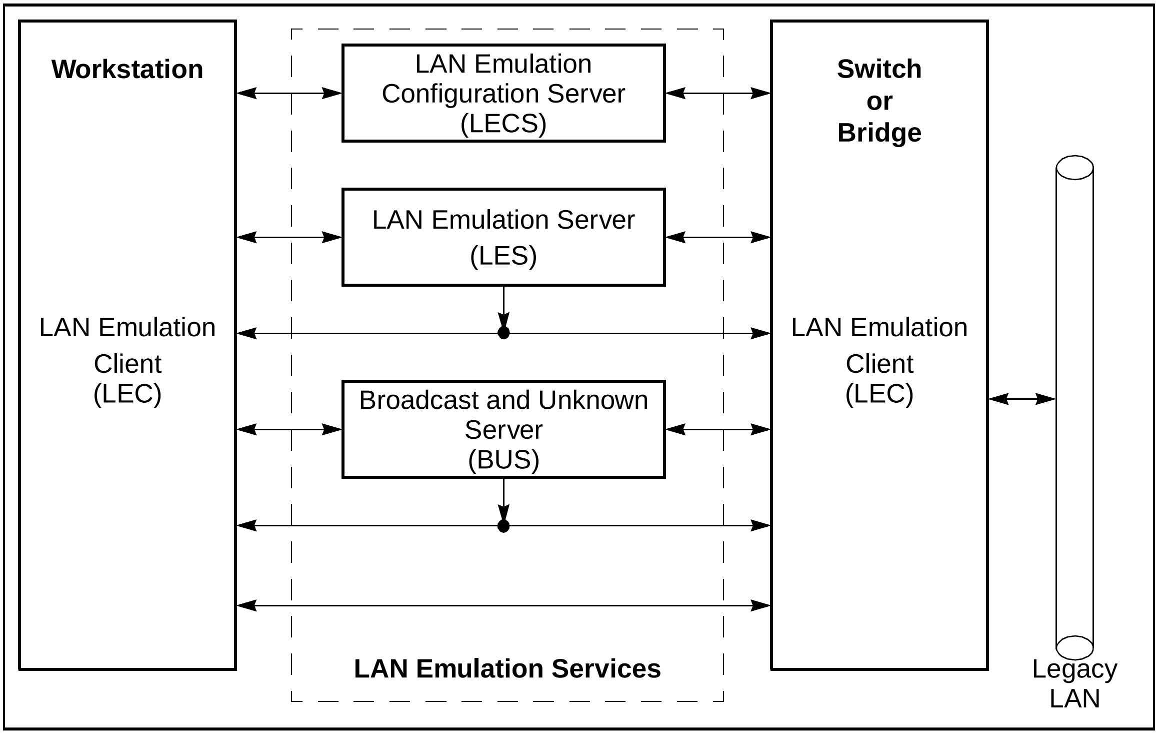

Figure 3.1 - Basic Emulated LAN Interconnections

3.2.1 LAN Emulation Client (LEC)

The LEC is the component in an end system that performs data forwarding, address resolution, and other control functions when communicating with other components within the ELAN. It also provides a MAC level emulated Ethernet or Token Ring interface and appears to higher level software as though a physical interface is present. Each LEC must register with both the LES and BUS associated with the ELAN it wishes to join before it may participate in the ELAN. To participate in multiple ELANs, an end system must have multiple LECs. Currently, up to 16 LECs are supported on ES-3810 switches and on adapter cards running Solaris or Windows/NT software, and up to 4 LECs on adapter cards running Windows/95.

3.2.2 LAN Emulation Configuration Server (LECS)

The LECS is responsible for the initial configuration of LECs. It provides information about available ELANs that a LEC may join, together with the address of the LES associated with each ELAN. Using DLE in *ForeThought* 5.0.x and greater, the user may also configure the LECS to associate multiple LES/BUS pairs with a given ELAN. This feature allows LECs to use a single, anycast address to reach one of the other DLE peer servers for their ELAN if their local server goes down. Normal address resolution through *ForeThought* PNNI, ATM Forum PNNI, or IISP will locate the closest, active LES which is using the anycast address.

3.2.3 LAN Emulation Server (LES)

The LES implements the control coordination function for the ELAN. The LES provides the service of registering and resolving MAC addresses to ATM addresses. A LEC registers its own address with the LES. A LEC also queries the LES when the client wishes to resolve a MAC address to an ATM address. The LES either responds directly to the client or forwards the query to other clients so they may respond. There may be more than one instance of an active LES per ELAN.

3.2.4 Broadcast and Unknown Server (BUS)

Unlike traditional shared-media LAN architectures such as Ethernet or Token Ring, ATM is connection based. Therefore, it has no built-in mechanism for handling connectionless traffic such as broadcasts, multicasts, and unknown unicasts. In an ELAN, the BUS is responsible for servicing these traffic types by accepting broadcast, multicast, and unknown unicast packets from the LECs via dedicated point-to-point connections, and forwarding the packets to all of the members of the ELAN using a single point-to-multipoint connection. (Unknown unicast packets are packets that the sending station broadcasts because it does not yet know the ATM address for the packet's destination MAC address. There may be more than one instance of an active BUS per ELAN. Using *ForeThought* 5.0.x and greater, each BUS must be a colocated BUS (also referred to as an intelligent BUS or a LES/BUS pair), which allows the BUS to use the LES's registration table to direct unicast traffic.

3.3 Emulated LAN Operation

This section describes the operation of an ELAN and its components from the point of view of a LEC. The operation of an ELAN may be divided into three phases:

1. Initialization
2. Registration and Address Resolution
3. Data Transfer

ELAN components communicate with each other using ATM connections. LECs maintain separate connections for traffic control functions and data transfer. The following connection types are used by the LEC when operating in an ELAN:

- *Configuration-Direct Connection*: a temporary bidirectional point-to-point VCC set up by the LEC to the LECS.
- *Control-Direct Connection:* a bidirectional point-to-point VCC set up by the LEC to the LES. This connection must be maintained for the duration of the LEC's participation in the ELAN.
- *Control-Distribute Connection*: a unidirectional point-to-multipoint VCC set up by the LES to the LEC. This connection must be maintained for the duration of the LEC's participation in the ELAN.
- *Multicast-Send Connection*: a bidirectional point-to-point VCC set up by the LEC to the BUS for sending multicast data to the BUS. The LEC must attempt to maintain this connection while participating in the ELAN.
- *Multicast-Forward Connection*: a unidirectional point-to-multipoint VCC set up from the BUS to LECs participating in the ELAN. The LEC must attempt to maintain this connection while participating in the ELAN.
- *Data-Direct Connection*: a bidirectional point-to-point VCC set up between LECs that want to exchange unicast data traffic, and torn down after 20 minutes (default) of inactivity. Each LEC normally establishes many Data-Direct Connections.

For the following discussion, please refer to Figure 3.2.

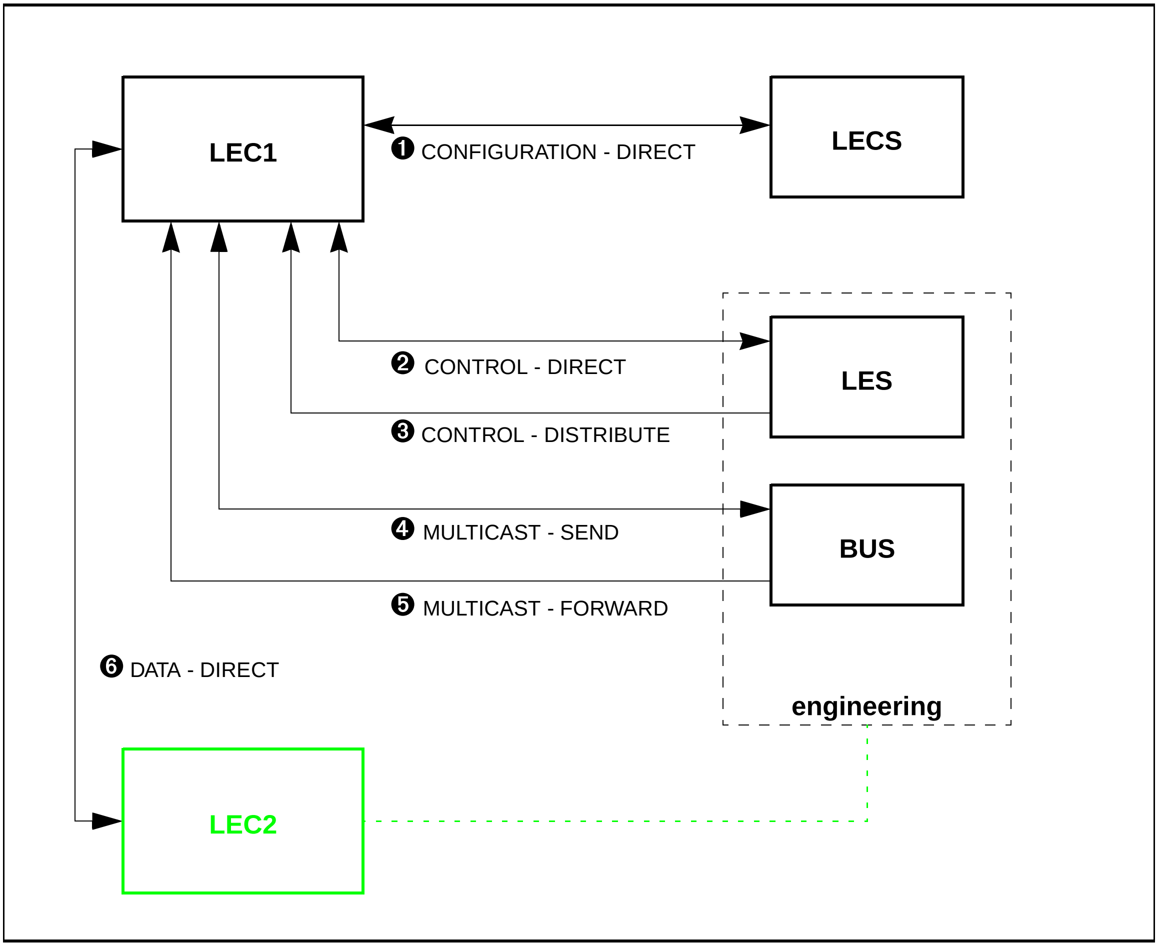

Figure 3.2 - ELAN Operation

3.3.1 Initialization

Upon initialization, LEC1 obtains its own ATM address via ILMI address registration. LEC1 obtains the address of the LECS in one of four ways: by querying the switch to which LEC1 is connected via ILMI, by connecting to the “well-known” address defined by the ATM Forum’s LANE standards (`47.0079.00.000000.0000.0000.0000.00A03E000001.00`), by using PVC (0,17), or by using an address that is locally configured on LEC1.

Once it knows the location of the LECS, LEC1 establishes a configuration-direct connection ❶ to the LECS. When connected, the LECS provides LEC1 with the information necessary to connect to the ELAN it wishes to join. This information includes such parameters as: the ATM address of the ELAN’s LES, the type of LAN being emulated, the maximum packet size, and the name of the ELAN (`engineering`, for example). This configuration information is contained in a configuration file that must be built and maintained by the network administrator.

NOTE

Detailed information about the LECS configuration file may be found in Section 3.6.1.

3.3.2 Registration and Address Resolution

After obtaining the address of the LES, LEC1 establishes a control-direct connection ❷ to the LES.

NOTE

When using DLE, this address is a single, anycast address which allows the LEC to reach one of the other DLE peer servers for its ELAN if its local server goes down. This address is routed via PNNI to the nearest active DLE peer server for this ELAN.

NOTE

If the LES is configured to perform ELAN access control (see Section 3.5), upon receiving a request from a LEC to join the ELAN, the LES sends a message to the LECS to verify that the LEC is allowed to join. If verification is received from the LECS, then the LES gives the LEC permission to join. If verification is not received from the LECS, the LES rejects the join request and the LEC is dropped.

The LES assigns LEC1 a unique identifier, and LEC1 registers its own MAC and ATM addresses with the LES. (The LES maintains a table containing the MAC addresses and corresponding ATM addresses of all members of the ELAN.) At this point, LEC1 has “joined” the ELAN.

The LES then establishes a control-distribute connection ❸ back to LEC1. Connections ❷ and ❸ can now be used by LEC1 to send LAN Emulation ARP (LE_ARP) requests to the LES, and receive replies.

LEC1 now sends an LE_ARP request to the LES to get the ATM address of the BUS corresponding to the broadcast MAC address (FF-FF-FF-FF-FF-FF). The LEC then establishes a multicast-send connection ❹ to the BUS. The BUS responds by setting up a multicast-forward connection ❺ to the LEC.

At this point, the LEC is ready to transfer data.

3.3.3 Data Transfer

When LEC1 receives a network-layer packet from a higher layer protocol to transmit to some destination MAC address (for example, LEC2), LEC1 initially does not know the corresponding ATM address of the destination. Consequently, LEC1 transmits an LE_ARP request to the LES.

NOTE

The example shown in Figure 3.2 assumes that LEC2 has already registered with the LES, and that connections similar to those described for LEC1 already exist.

While waiting for the LES to respond, LEC1 forwards the packet to the BUS. The BUS broadcasts the packet to all LECs on the ELAN. This is done to avoid data loss, and to minimize connection set-up latency (due to the LE_ARP process) that may not be acceptable to some network protocols.

If the LE_ARP response is received, LEC1 establishes a data-direct connection ❻ to the destination address of LEC2. This path will be used for subsequent data transfers. Before LEC1 begins to use this connection, it first sends a “flush” packet via the BUS to the destination, LEC2. When LEC2 acknowledges receipt of this packet, signifying that the BUS path is empty, only then does LEC1 begin to use the data-direct connection ❻ for data transfer. This process ensures that the network protocol’s frames arrive in the proper order.

If no response is received to the LE_ARP, LEC1 continues to send data via the BUS, while continuing to LE_ARP until a response is received and a data-direct connection to LEC2 is established.

If LEC1 already has a data-direct connection to a MAC address it wishes to reach, it need not go through the LE_ARP process again. Instead, it continues to use the current connection. This is possible because each LEC maintains a cache of MAC address to ATM address mappings that it receives in response to the LE_ARPs it has sent. Entries in this cache are “aged” out over a period of time. Data-direct connections are also cleared if they remain inactive for a period of time.

3.4 Distributed LAN Emulation

Distributed LAN Emulation (DLE) allows the LES and BUS functions that are provided to each ELAN to be distributed among multiple, interconnected server platforms. In this way, DLE provides these ELANs with resiliency and scalability.

To understand DLE operation, it is useful to compare DLE to the current LANE service model, which uses a single LES and BUS for each ELAN. This section first describes a simple example of the single server model and then gives a detailed overview of the DLE model.

3.4.1 Single Server LANE Services Model

Figure 3.3 shows the topology of a single server supporting an ELAN. In this example, the LECs are hosts that are using IP, and the LES and BUS are running on the same switch. Three LANE LECs are all registered in the same ELAN called Eng, and each is, therefore, connected to a LES and to a BUS for that ELAN.

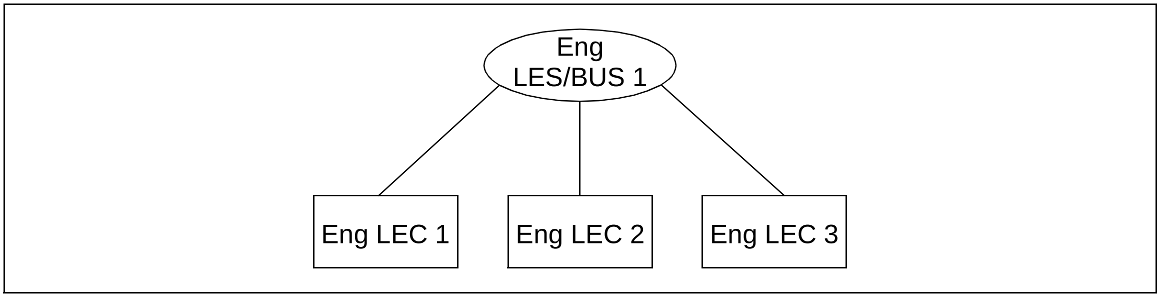

Figure 3.3 - Single Server LANE Services Model

3.4.1.1 Using a Single Server

When LEC 1 wants to contact LEC 3, several messages are exchanged. First, LEC 1 attempts to learn the MAC address of LEC 3 by broadcasting an IP-ARP request with LEC 3's IP address. As Figure 3.4 shows, this ARP request is sent in two steps: ❶ as a point-to-point message from LEC 1 to the LANE BUS, then ❷ as a point-to-multipoint message from the BUS to all of the LECs registered in the ELAN.

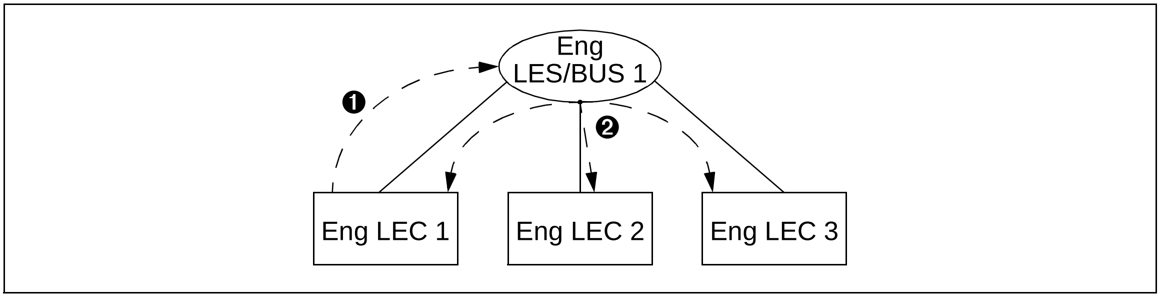

Figure 3.4 - Broadcast IP-ARP Request

When LEC 3 receives the IP ARP request, it recognizes that it is the intended destination, and, therefore, attempts to send an IP ARP response to LEC 1 (whose MAC address was supplied in the ARP request packet).

As shown in Figure 3.5, the delivery of the ARP response is a three-step process: ❸ LEC 3 sends an LE-ARP query to the LES, asking for the ATM address that corresponds to LEC 1's MAC address; ❹ the LES sends an LE-ARP response to LEC 3; and ❺ LEC 3 establishes a circuit to LEC 1's ATM address.

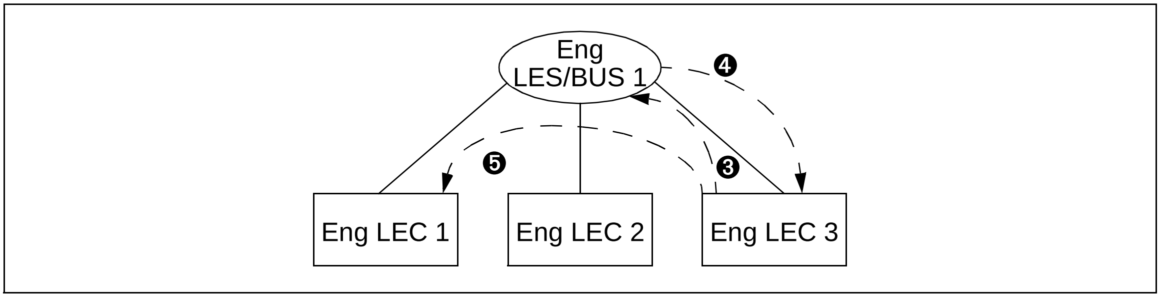

Figure 3.5 - IP ARP Response Handling

3.4.1.2 Limitations of a Single Server

Because the there is only one LES/BUS supporting the ELAN, the following limitations exist:

- The number of LECs in a single ELAN is limited by the number of virtual circuits that the single LES/BUS can establish through their platform's ATM port. This usually limits the ELAN to about 500 LECs.
- Clusters of LECs that are geographically separated from the LES/BUS may have poor throughput, even when connecting to each other, because address queries and broadcasts may traverse slow wide-area links.
- A failure of the LES or BUS brings down the ELAN.

3.4.2 Distributed LAN Emulation Model

To address the limitations of the single server model, DLE distributes the LANE services load among a mesh of LES/BUS DLE peer servers, as shown in Figure 3.6.

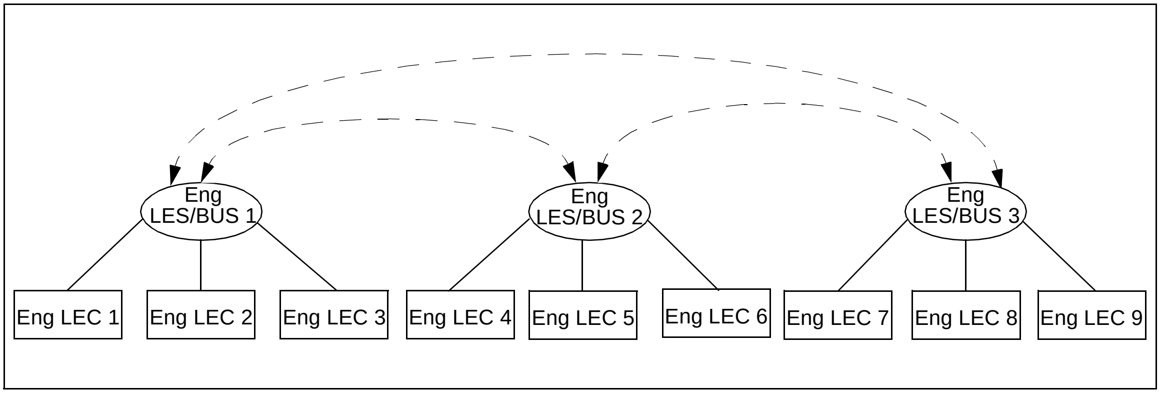

Figure 3.6 - Distributed LAN Emulation Model

Each DLE peer server actually maintains two sets of connections: one is a point-to-multipoint connection to each of its peers for broadcasting multicast data and flooding control information, and the other includes individual point-to-point connections to each peer for directed control traffic.

Each DLE peer server that supports the ELAN is responsible for registering and giving reports about the LECs that are attached to it directly. Each DLE peer server propagates this information to both its locally attached LECs and its peers.

NOTE

Each device running a DLE peer server must use *ForeThought* 5.0 or greater; however, the DLE peer servers support clients and attached switches using *ForeThought* 4.0 and 4.1, and third-party devices that are ATM Forum LANE 1.0 compliant.

3.4.2.1 Using DLE

Figure 3.7 shows how a connection begins to be established through DLE peer servers. LEC 1 wants to communicate with LEC 9, which is in the same ELAN, but is locally attached to a different DLE peer server. First, ❶ LEC 1 sends an IP ARP broadcast request to its local DLE BUS. Then, ❷ the BUS broadcasts the packet to both its locally attached LECs and its DLE peer servers.

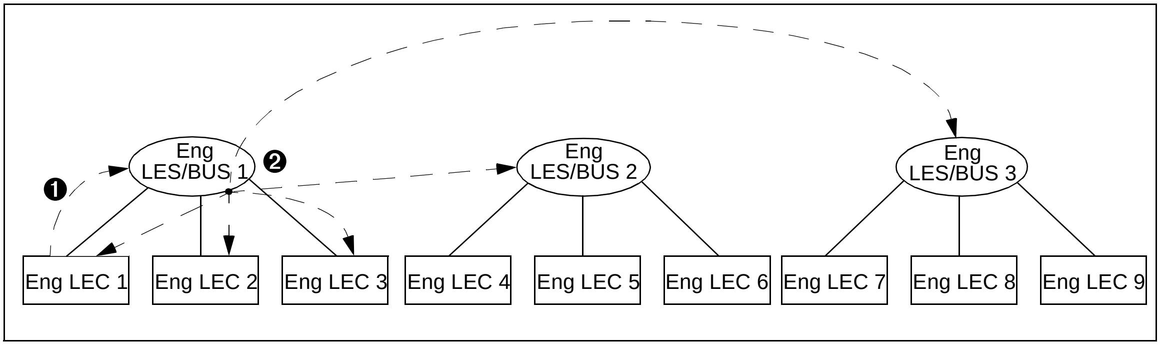

Figure 3.7 - IP ARP Broadcast from LEC 1 to LEC 9

Upon receiving the broadcast from the first DLE peer server, the peers re-distribute the packet to their own locally attached LECs ❸, as shown in Figure 3.8, so the packet arrives its actual destination at LEC 9.

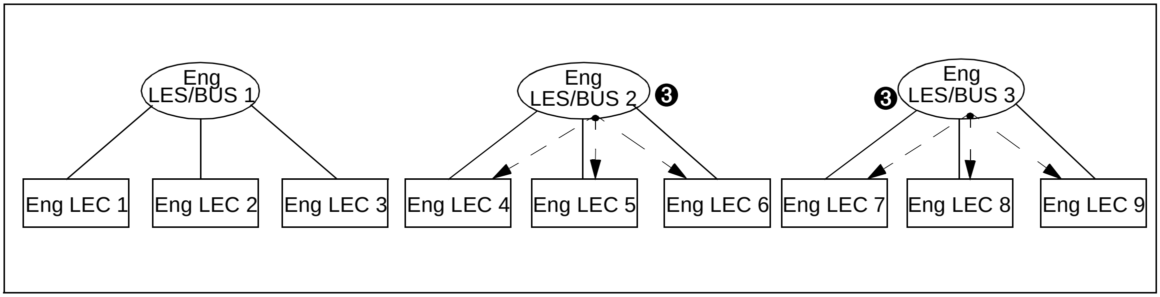

Figure 3.8 - Re-distributing the Broadcast across DLE Peer Servers

NOTE The peers do not re-distribute the packet to other peers; this would create a loop.

LEC 9 recognizes its IP address, and prepares an IP ARP response. As shown in Figure 3.9, it then sends an LE-ARP request to its local LES ❹, asking for the ATM address that matches LEC 1's MAC address. Since LEC 9's local LES does not have an entry for LEC 1, the local LES passes the query along to all of its locally-attached proxy LECs (none are shown in this figure) and all of its DLE peer servers ❺.

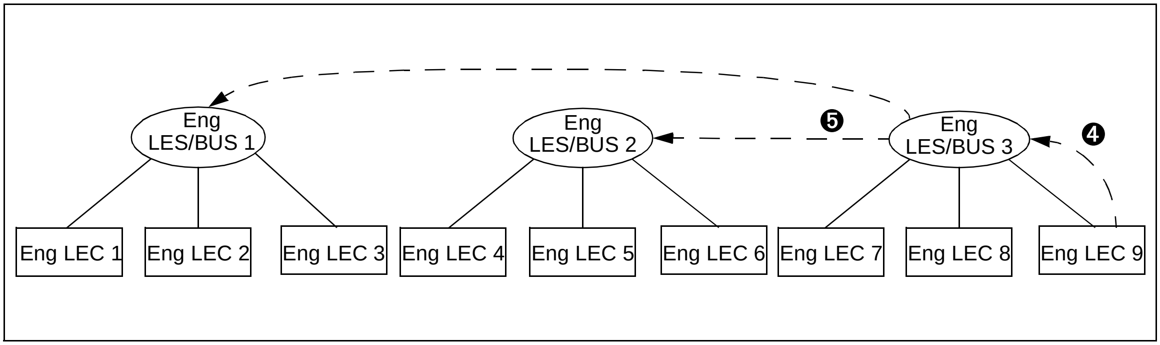

Figure 3.9 - LE-ARP for Unknown Host Sent to Proxies (not shown) and DLE Peer Servers

In Figure 3.10, the second DLE peer server is attached to two proxy LECs (LEC 4 and LEC 5). When the DLE peer server receives the LE-ARP query, it cannot resolve the query, so the DLE peer server re-distributes the query to its proxy LECs ❻ (but not to its peer servers again, to avoid a loop). Meanwhile, the first peer server has been able to resolve the LE-ARP for the address of LEC 1 and has sent an LE-ARP response to the third server ❼.

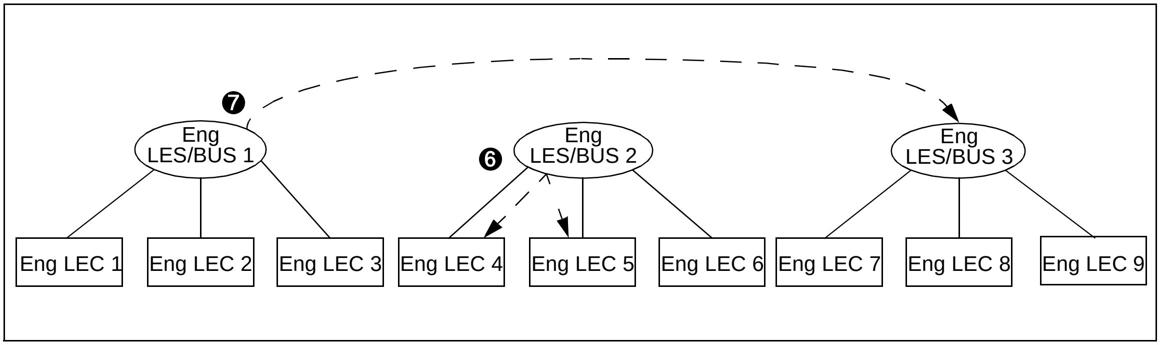

Figure 3.10 - LE-ARP Query Answered by One DLE Peer Server and Re-distributed by Another

When the third DLE peer server receives the LE-ARP response, it passes it directly to LEC 9 (which sent the original query) ❽. The third DLE peer server also caches the registration information for LEC 1 so that other local LECs do not have to go through the entire process again. However, this cache ages out over time. LEC 9 can now open a connection to LEC 1, and send its IP ARP response ❾, as shown in Figure 3.11.

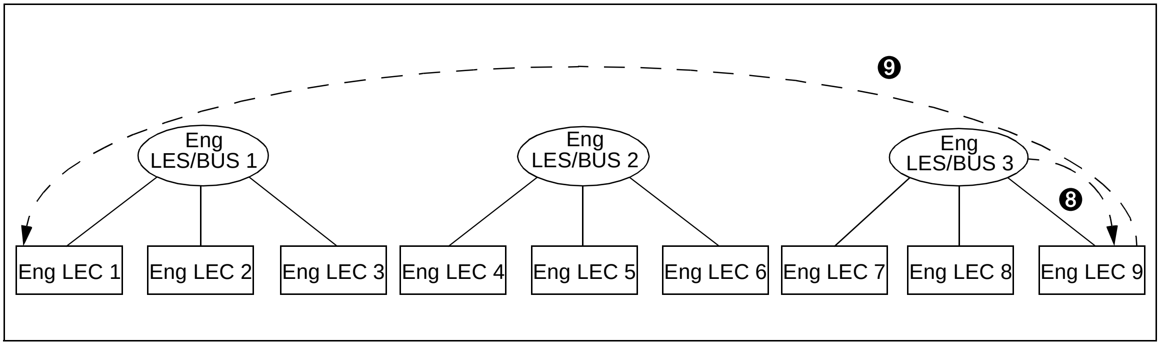

Figure 3.11 - LE-ARP Response Delivered and LEC 9 Contacts LEC 1

3.4.2.2 Advantages of DLE

As mentioned earlier, using DLE provides solutions to the problems of using a single server.

3.4.2.2.1 Load Sharing

DLE peer servers distribute the circuit and processing load. The number of LANE LECs is no longer limited by the number of circuits one LES/BUS platform can maintain, since many platforms can support a single ELAN. Also, more VCs are available for use by other applications.

3.4.2.2.2 Improved Performance for Remote LECs

With DLE, broadcast delivery and LE-ARP resolution across peer servers can take a little longer than if all LECs were connected to a single server, since extra processing steps and transmissions are needed. However, ELANs with groups of LECs in different locations can be designed for higher performance by providing a DLE peer server with each group. Broadcasts and address resolution within each group will improve.

3.4.2.2.3 Fault Tolerance

Perhaps the most important advantage of DLE is fault tolerance. In a single server ELAN, the server can be a single point of failure. If the server fails, endstations in the ELAN are unable to discover each other through broadcast queries and unable to resolve MAC addresses into ATM addresses. Increased network reliability, therefore, requires that ELANs have backups for LES and BUS functions. To illustrate this point, the single server model is again discussed.

3.4.2.2.3.1 Single Server ELAN

Figure 3.12 shows a single server ELAN composed of nine LECs attached to three different switches. The LECS and the LES/BUS are attached to a host connected to a single switch. The process for LEC 1 to connect to the LANE services takes several steps:

1. LEC 1 asks the signalling software on its switch to open a connection to the “well-known” LECS address. (Other addressing methods could also be used).
2. The signalling software knows that this address is attached to port N (the port on which the host resides on the switch), and opens a circuit between LEC 1 and port N.
3. LEC 1 sends a message to the LECS, asking for the address of the LES for LEC 1’s ELAN. The LECS responds with the ATM address of the LES, and LEC 1 establishes a circuit to the LES and then the BUS.

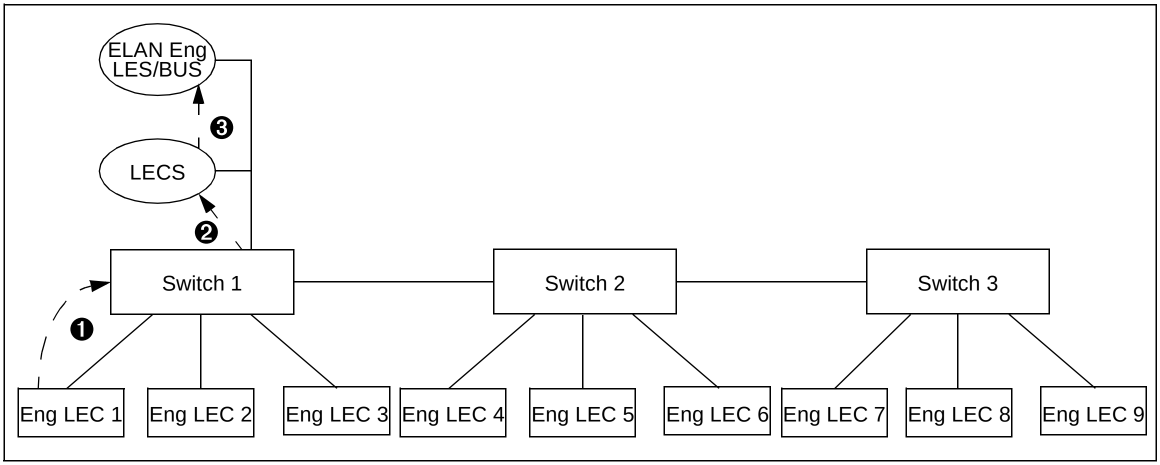

Figure 3.12 - ELAN with a Single Server and Multiple Switches Connecting to Services

In Figure 3.13, when LEC 7 goes through the same process, it is slightly more complicated. When LEC 7 asks its local switch’s signalling software to establish a circuit to the LECS ❶, the local switch must use inter-switch link information (IISP or PNNI tables) to establish a cross-switch circuit to the LECS. Once this circuit is established, however, the process is identical.

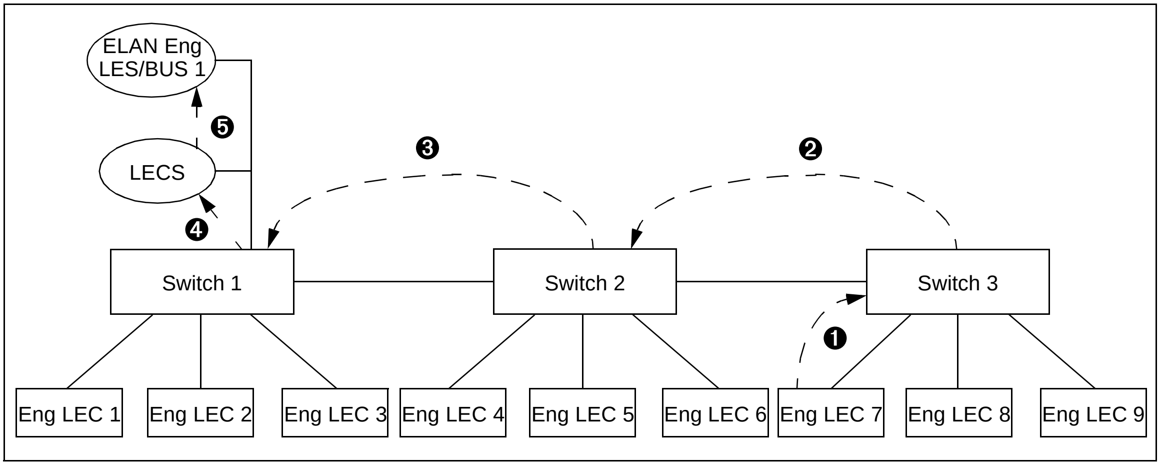

Figure 3.13 - ELAN with Single Server and Remote Connection to Server

Figure 3.14 shows the ELAN in operation after three LECs have gone through the registration process.

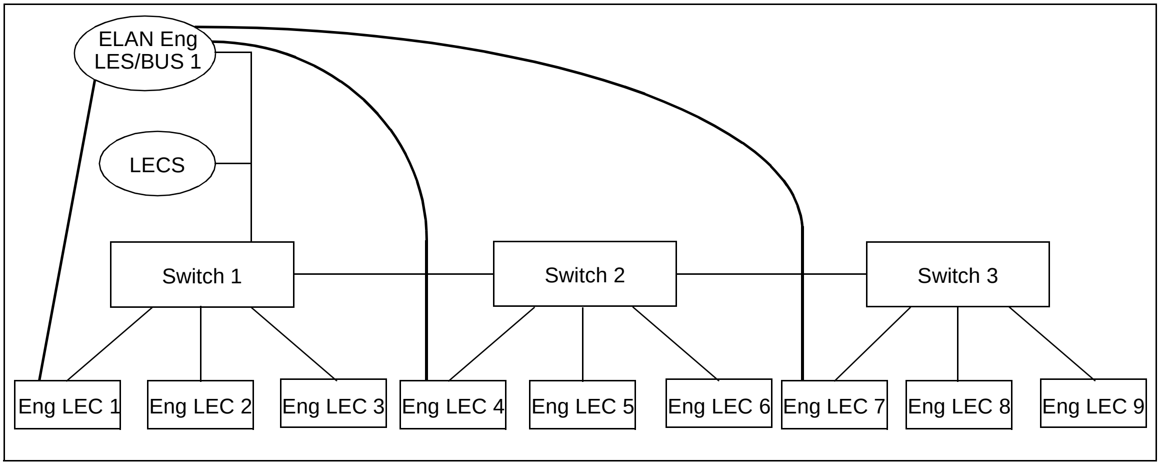

Figure 3.14 - ELAN with Single Server in Operation

If the single server in Figure 3.14 goes down, then the entire ELAN goes down. At this point, the administrator must intervene and reconfigure the ELAN.

3.4.2.2.3.2 DLE ELAN

As noted previously, having a single server supporting an ELAN has a potential problem because the server can be a single point of failure. However, DLE can address this problem. By attaching the ELAN LECs to multiple DLE peer servers which communicate with each other as described earlier, the number of LECs affected by a server failure is reduced, and a backup server is provided for affected LECs to use. Figure 3.15 shows the configuration of such an ELAN as three stations register.

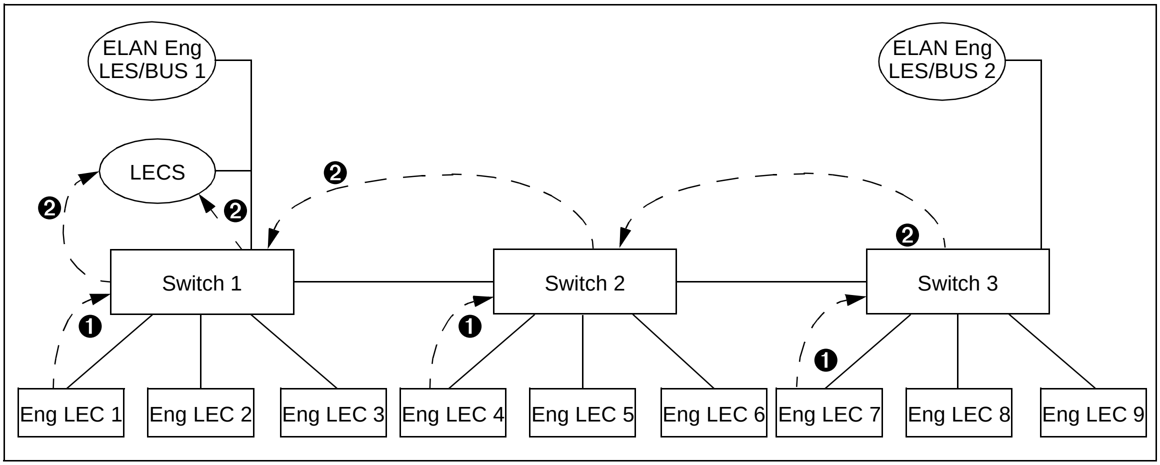

Figure 3.15 - Registrations on an ELAN with Multiple Servers

LECs 1, 4, and 7 are directed by their switches to the LECS. The result is shown in Figure 3.16.

NOTE

The connection between the two servers carries broadcasts and LE-ARP traffic as described earlier.

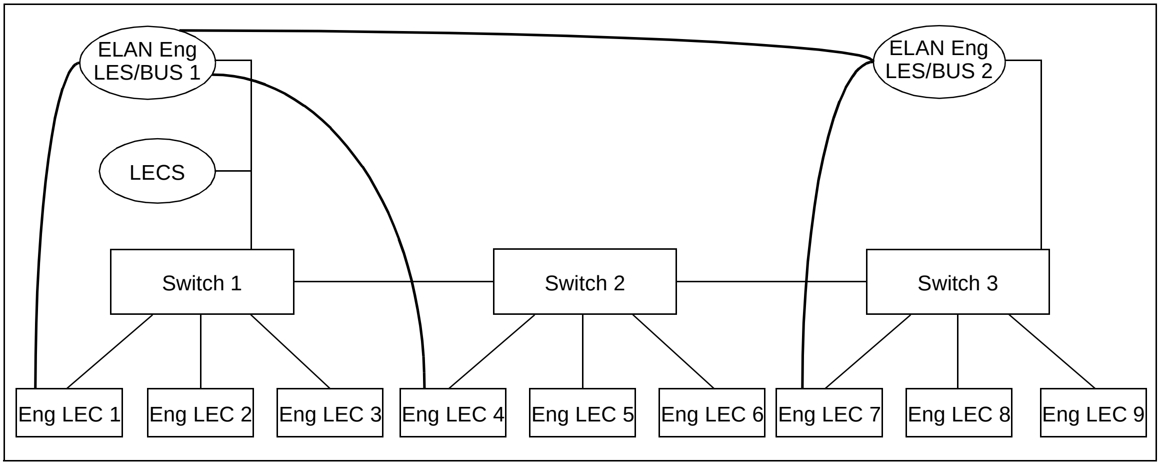

Figure 3.16 - ELAN with Multiple Servers in Operation

This ELAN may experience significant performance improvements for the reasons described earlier. Even if the actual performance is similar to using a single server in a particular network, a great advantage is gained through its fault-tolerance if one of the servers fails as depicted in Figure 3.17.

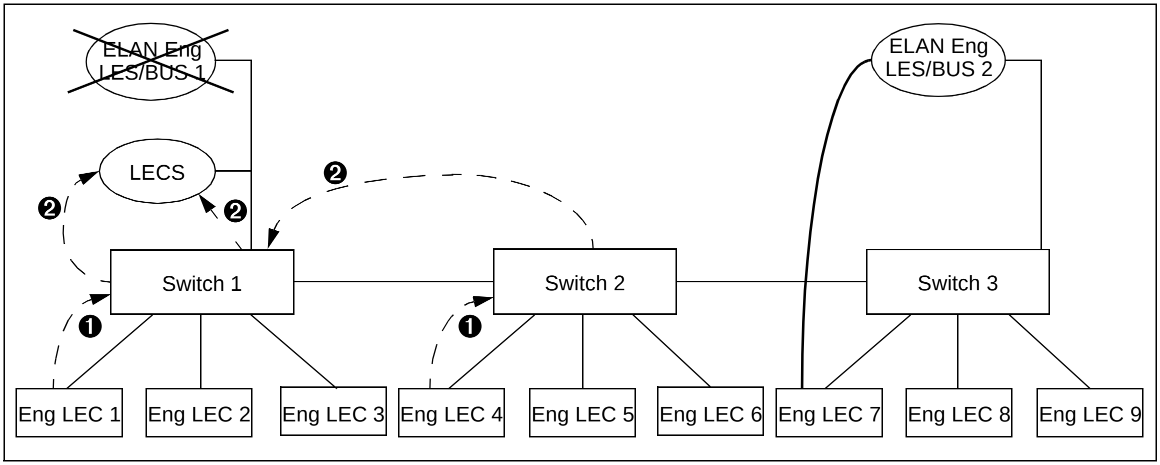

Figure 3.17 - Failure of One ELAN Server and the Recovery Process

The failure and recovery process occurs as follows:

1. Eng LES/BUS 1 has lost power. All circuits connected to it are torn down. Low-level signalling traffic (e.g., SSCOP messages) stop, and Switch 1 removes the address of Eng LES/BUS 1 from its link tables.
2. LECs 1 and 4 had been connected to Switch 1. They detect that their connections to Eng LES/BUS 1 have been torn down; user intervention is not necessary.
3. LECs 1 and 4 follow LANE 1.0 protocols to locate an LECS again to find the address of their ELAN's LES. In this example, they again connect to the LECS.
4. The LECS reports to LECs 1 and 4 that their ELAN server is at ATM address N1. This address is used by every peer LES supporting the ELAN; both Eng LES 1 and Eng LES 2 in this example.
5. LEC 1 sends a request to Switch 1 to establish a connection to address N1. Switch 1 no longer believes it has a direct connection to N1, and instead uses PNNI to establish a circuit through Switches 2 and 3 to Eng LES 2.
6. LEC 4 sends a request to Switch 2 to establish a connection to N1. Switch 2 may have learned from Switch 1 that it no longer offers a connection to N1, or Switch 2 may attempt a route through Switch 1 and be "bounced back" through ATM Forum PNNI crankback. Either way, Switch 2 finally routes the connection through Switch 3 to Eng LES 2.

This recovery process occurs quickly -- clients typically recover at a rate of 100 clients per minute -- and the result is a re-configured ELAN as shown in Figure 3.18.

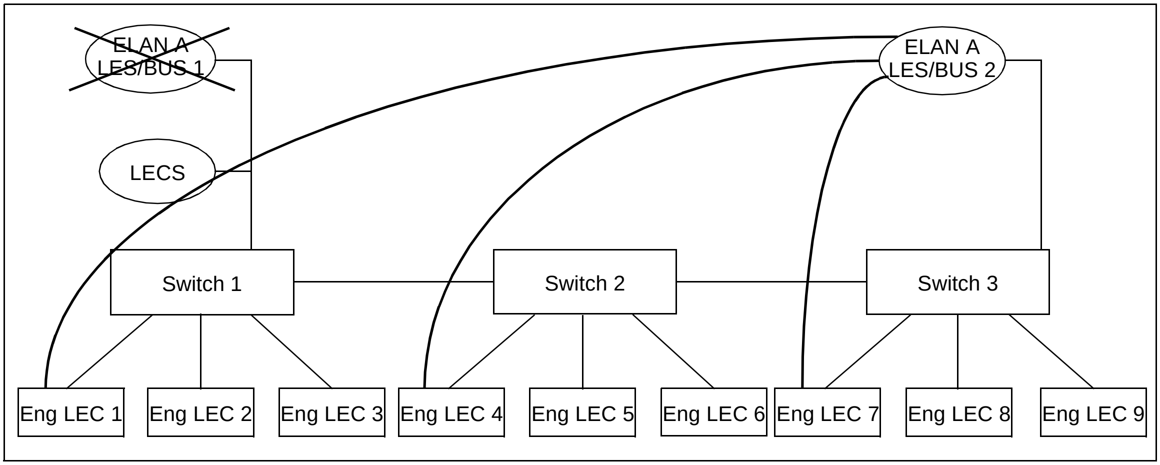

Figure 3.18 - ELAN Re-established Using the Second Server

3.5 ELAN Access Control

Basic ATM Forum LAN Emulation Servers do not guard against unauthorized users learning an ELAN's LES address and then joining the ELAN. However, a method of authorization checking is available. After a LEC obtains the address of its LES, the LEC sends a request to the LES to join the ELAN. If the LES has ELAN access control enabled, it sends a message to the LECS to verify that the LEC is allowed to join. If verification is received from the LECS, then the LES gives the LEC permission to join. If verification is not received from the LECS, the LES rejects the join request and the LEC is dropped.

Using this feature, an authorization check is also performed each time the LECS reloads the LECS configuration file. (The LECS periodically checks whether its configuration file has been modified, and, if it has, the file is re-read. The length of this period, in seconds, is defined by the `Reload_Period` key.) If the file has changed to disallow some clients that were previously allowed, those clients will be dropped from the ELAN.

NOTE

ELAN access control also works with a third-party LECS. The LES revalidates the client every 600 seconds since the third-party LECS will not contact the LES with configuration changes.

You can enable ELAN access control when you are creating the LES. When you use the **conf lane les new** command, specify the **-secure** option. This indicates you want to activate a secure LES/BUS pair. (ELAN access control is disabled by default.)

```
myswitch::configuration lane les> new <LES Selector Byte (HEX)> <LES name> \
        [-bus <BUS Selector Byte (HEX)>] \
        [-type (ethernet | token-ring)] \
        [-mtu (1516 | 1580 | 4544 | 9234 | 18190)] \
        [-secure wka | <LECS ATM Address>] \
        [-registertlvs (enable | disable)] \
        [-elanid <ELAN ID>] \
        [-segid <TR Segment ID>] \
        [-fwdarp (enable | disable)] \
        [-anycast <LES Anycast ATM Address>] \
        [-peers <atm-addr> ...]
```

If you enter **wka** with the **-secure** option, the ATM Forum well-known LECS address is used. In this case, you do not have to type the actual well-known address. However, if you are using an LECS address that is different than the well-known address, then you must type the full LECS ATM address to be used.

If you want to disable ELAN access control, or if you want to enable ELAN access control at a later time after the LES has been created, you can use the **conf lane les security** command to do this. See Part 1 of the *AMI Configuration Commands Reference Manual* for more information. By using this command, you do not have to delete and recreate the LES.

3.6 Configuring an ELAN

NOTE

When configuring LANE services in your network, it is highly recommended that you use a switch that has a 32MB SCP or greater to run those services.

There are different instructions for configuring an ELAN, depending on how your network is currently configured. Please read the following list to determine which set of instructions to use.

- If you had previously configured LANE, you want to upgrade some or all of the clients to *ForeThought* 6.x, and you want to upgrade all the equipment that is running services to *ForeThought* 6.x using DLE, use the instructions found in Section 3.7.
- If you had previously configured LANE, you want to leave the clients running *ForeThought* 4.1.x, and you want to upgrade all the equipment that is running services to *ForeThought* 6.x **without** using DLE, use the instructions found in Section 3.8.
- If you are configuring LANE for the first time and all of your equipment is running *ForeThought* 6.x, use the instructions that follow here in Section 3.6.

To configure an ELAN on a switch, you must log into AMI on a switch running *ForeThought* 6.0 and use the commands found under **`configuration lane`**.

NOTE

More information about each of these commands may be found in Part 1 of the *AMI Configuration Commands Reference Manual*.

There are three major steps that the system administrator should follow in order to configure and maintain ELANs:

1. Configure an LECS configuration database file.
2. Start the LAN Emulation Services (LECS and LES/ BUS).
3. Start the LEC(s) and join an ELAN.

NOTE

Steps 1 may be performed using a text editor on any system. However, the resulting file can be **used** only on systems running under Solaris 2.5, 2.5.1, or 2.6, or on a FORE switch running *ForeThought* 4.0 or greater or a *TNX* switch running *ForeThought* 5.1 or greater.

The remainder of this section gives a practical example of configuring and administering an ELAN.

3.6.1 Configuring an LECS Configuration Database File

The LECS uses a text configuration file to contain the configuration information needed by LECs that wish to participate in an ELAN. The LECS configuration file may be built and edited using a text editor such as `vi` or `emacs`.

NOTE

ForeThought VLAN Manager, a product available separately from FORE Systems, greatly simplifies the administration of ELANs. Its graphical user interface controls the content of the LECS configuration file transparently to the user. For more information, please refer to the *ForeView VLAN Manager User's Manual*.

3.6.1.1 Before You Begin

Before building or modifying the LECS configuration file, you should first determine the topology of the ELAN or ELANs that you want to administer. You must supply the following information when building or editing the LECS configuration file:

- Provide the name of each ELAN (`engineering`, `marketing`, etc.).
- Provide the LAN type (Ethernet or Token Ring) for each ELAN.
- Provide the MTU size for each ELAN.
- Provide the ATM address of the LES for each ELAN. If you are using DLE, this address must be the anycast address for the DLE peer servers in each ELAN. Be sure to choose a distinct anycast address for each ELAN in the network. It must be unique within the first 19 bytes.
- Provide the address of each LEC that may participate in each ELAN.
- Provide the MPOA control parameters if you wish to run MPOA.
- If you wish LECs to use a default ELAN, the default LES information must also be included.
- Provide various other configurable parameters.

CAUTION

Do not attempt to edit an existing functional LECS configuration file without first making a backup copy of the file. Incorrect modification of the configuration file could result in loss of communication between one or more members of a defined ELAN.

Configuring an Emulated LAN

> **NOTE**
>
> You may make changes to the LECS configuration file while the LECS process is running. The configuration file is reread periodically by the LECS process (the default period is ten minutes). Consequently, any changes that you make to the configuration file are not recognized until the file is reread.

3.6.1.2 LECS Configuration File Syntax

Each line that you enter in the configuration file takes the general form:

```
[[group].]key : value
```

The `group` field may represent:

- ELANs (by name) - ELAN names are case-sensitive, and may not exceed 32 characters in length
- clients ATM or MAC addresses
- miscellaneous LECS control information specified by using a `group` name of `LECS`

The `key` field is used to denote an individual parameter within a `group`.

The `value` field contains the value assigned to the `key`.

Omitting the `group` implies that the `key` and `value` apply to all groups in the configuration file. Leading and trailing spaces, as well as spaces on either side of the “:”, are ignored.

For example, to specify a maximum frame size of 1516 bytes for the ELAN named `engineering`, enter the following:

```
engineering.Maximum_Frame_Size : 1516
```

Similarly, to specify a default maximum frame size of 1516 bytes for all ELANs defined in a given configuration file, enter the following:

```
.Maximum_Frame_Size : 1516
```

Table 3.1 defines the various key parameters that may be entered in the configuration file. The acceptable range of values and the default value for each parameter is also given.

Table 3.1 - LECS Configuration File Parameters

Parameter	Definition
`.LAN_Type: Ethernet/IEEE 802.3`	Identifies the type of ELAN, either Ethernet/IEEE 802.3 or IEEE 802.5. The default is Ethernet/IEEE 802.3.
`.Maximum_Frame_Size: 1516`	Specifies the length (in number of bytes) of the largest frame. Selections are: 1516, 1580, 4544, 9234, and 18190. The default is 1516 for Ethernet and 4544 for Token Ring.
`.Control_TimeOut: 120`	Specifies the timing out of request/response control frame interactions, in seconds. The minimum is 10 seconds and the maximum is 300 seconds. The default is 120 seconds.
`.Maximum_Unknown_Frame_Count: 1`	Limits the number of unicast frames sent to the BUS. The minimum is 1 frame per 60 seconds and the maximum is 10 frames per second. The default is 1 frame per second.
`.Maximum_Unknown_Frame_Time: 1`	Limits the number of unicast frames sent to the BUS in the specified number of seconds. The default is 1 second.
`.VCC_TimeOut_Period: 1200`	Specifies the length of time that an idle data connection remains open before being closed. The default value is 1200 seconds.
`.Maximum_Retry_Count: 1`	Limits the number of LE_ARP retransmission requests. The minimum is 0 and the maximum is 2. The default is 1.
`.Aging_Time: 300`	Specifies the period that LE_ARP cache table entries remain valid, in seconds. The minimum is 10 and the maximum is 300. The default value is 300 seconds.
`.Forward_Delay_Time: 15`	Specifies the timing out of non-local ARP cache entries in seconds. The minimum is 4 and the maximum is 30. The default value is 15 seconds.
`.Expected_LE_ARP_Response_Time: 1`	Specifies the maximum time a LEC expects an LE_ARP request/response will take, in seconds. The minimum is 1 and the maximum is 30. The default value is 1 second.
`.Flush_TimeOut: 4`	Specifies the maximum time a LEC expects an LE_FLUSH request/response will take, in seconds. The minimum is 1 and the maximum is 4. The default value is 4 seconds.

Table 3.1 - LECS Configuration File Parameters (Continued)

Parameter	Definition
`.Path_Switching_Delay: 6`	Specifies the minimum time between switching BUS and data paths, in seconds. The minimum is 1 and the maximum is 8. The default value is 6 seconds.
`.Local_Segment_ID`	The segment ID of the emulated LAN for an IEEE 802.5 source routing bridge.
`.ELAN_Identifier: auto`	Specifies a non-zero ELAN identifier. The default is auto, which signals the LES to generate an ELAN identifier itself.
`.Multicast_Send_VCC_Type: Best Effort`	Specifies the multicast send mode, either Best Effort, Variable, or Constant. The default is Best Effort.
`.Connection_Complete_Timer: 4`	Specifies the time period in which data or READY_IND is expected, in seconds. The minimum is 1 and the maximum is 10. The default is 4 seconds.
`.ShortCut_Protocols: IP`	Specifies the set of protocols on which to perform flow detection. Also, specifies the set of protocols for which MPOA resolution is supported. The default is {}. Currently, the only supported value is IP.
`.ShortCut_Threshold: 10/1`	Specifies the number of frames per second that a LANE/MPOA client (LEC/MPC) forwards to the same destination via the default forwarding path before which it should begin using a shortcut. The minimum number of frames is 1 frame and the maximum is 65,535 frames. The minimum rate of speed at which the frames are forwarded is 1 second and the maximum is 60 seconds. The default is 10 frames per second.
`.Resolution_Initial_Retry_Time: 5`	Specifies the initial retry time interval after which a LEC/MPC may send another MPOA Resolution Request if an MPOA Resolution Reply has not been received for the initial request. The minimum is 1 second. The maximum is 300 seconds. The default is 5 seconds.
`.Resolution_Maximum_Retry_Time: 40`	Specifies the maximum retry time interval after which a LEC/MPC assumes an MPOA Resolution Request has failed. The minimum is 10 seconds. The maximum is 300 seconds. The default is 40 seconds.

Table 3.1 - LECS Configuration File Parameters (Continued)

Parameter	Definition
`.Resolution_Hold_Down_Time: 160`	Specifies the minimum amount of time to wait before re-initiating an MPOA Resolution Request after a failed resolution attempt. This value is usually greater than the Resolution_Maximum_Retry_Time. The minimum is 30 seconds. The maximum is 1,200 seconds. The default is 160 seconds.
`.MPOA_KeepAlive_Time: 10`	Specifies how often an MPS must send MPOA Keep-Alive messages to all LEC/MPCs for which it has created and is maintaining Egress Cache Entries. The minimum is 1 second. The maximum is 300 seconds. The default is 10 seconds.
`.MPOA_KeepAlive_Lifetime: 35`	Specifies the length of time that a Keep-Alive message is considered valid. It is recommended that this value be at least twice the value of the MPOA_KeepAlive_Time. The minimum is 3 seconds. The maximum is 1,000 seconds. The default is 35 seconds.
`.Resolution_GiveUp_Time: 40`	Specifies the minimum amount of time to wait before an MPS should give up on a pending resolution request. The minimum is 5 seconds. The maximum is 300 seconds. The default is 40 seconds.
`.Resolution_Default_Holding_Time: 1200`	Specifies the default holding time for use in NHRP Resolution Replies. The minimum is 60 seconds. The maximum is 7,200 seconds. The default is 1,200 seconds.
`.IP_Flows: <ip-flow> {,<ip-flow>}`	Specifies a set of IP flows that are defined by the parameters listed below.
`<ip-flow>`	Specifies the IP information that is defined by the parameters listed below. The format is `<src-dst> [<proto>] <thresholds> <qos>`.
`<src-dst>`	Specifies the source and destination addresses of an IP flow. The format is `<ip-addr> <ip-addr>` with the source address first and the destination address second.
`<ip-addr>`	Specifies the format for the IP addresses `<ip-addr>` used above. The format is `<dotted-addr>/<prefix-leng>`.

Table 3.1 - LECS Configuration File Parameters (Continued)

Parameter	Definition
`<dotted-addr>`	Specifies the format for the IP addresses `<dotted-addr>` used above. The format is `<octet>.<octet>.<octet>.<octet>`. The range of each octet is from 0 to 255.
`<prefix-leng>`	Specifies the prefix length to indicates how much of a given IP address is significant. The range is 0 through 32. 0 means any IP address can be matched and 32 means the entire IP address is significant.
`<proto>`	Optionally specifies for which protocol this flow applies (ICMP, IGMP, TCP, or UDP). The format is one of: `ICMP`, `IGMP`, `TCP <port> <port>`, or `UDP <port> <port>`. If no protocol is specified, then this flow applies to all protocols.
`<port>`	Specifies the source and destination port number, respectively, to be used with the `<proto>` parameter. 0 indicates a match for any port number. The range is 0 through 65535.
`<thresholds>`	Specifies a traffic threshold that must be reached before a shortcut VCC is established for a flow. Also, specifies a threshold below which a shortcut VCC is no longer considered valid, is torn down, and the default forwarding path is used. Both thresholds are expressed in frames per second. The format is `<threshold> <threshold> <threshold>` with the set-up threshold first, the tear-down threshold second, and the set-up threshold for the default forwarding path third.
`<threshold>`	Specifies the format for entering the threshold data as described for `<thresholds>`. The format is `<frame-count>/<frame-time>`. For set-up, a zero count with a non-zero time means a shortcut VCC should be established upon the first frame and a non-zero count with a zero time means a shortcut VCC is never established. For tear-down, a zero count with a non-zero time means a shortcut VCC is never torn down.
`<frame-count>`	Specifies the format for the number of frames for the set-up and tear-down thresholds. The range is 0 through 65535.

Table 3.1 - LECS Configuration File Parameters (Continued)

Parameter	Definition
`<frame-time>`	Specifies the format for the number of seconds for the set-up and tear-down thresholds. The range is 0 through 65535.
`<qos>`	Specifies the parameters used in signalling a connection for the specified flow. The format is `<qos-flags>` `<qos-class>`. Currently, `<qos-flags>` is unused and its value should be 0.
`<qos-class>`	Specifies the Quality of Service (QoS) class for this flow. `Shared` specifies a UBR connection will be shared among all flows to the same destination. `UBR` specifies a non-shared UBR connection. The format is `UBR <rate>`. `CBR` specifies a CBR connection. The format is `CBR <rate>`. `VBR` specifies a VBR connection. The format is `VBR <rate> <rate> <burst>` (for CLP=0+1 cells) `<rate> <burst>` (for CLP=0 cells). `NRT-VBR` specifies a non-real time VBR connection. The format is `NRT-VBR <rate> <rate> <burst>` (for CLP=0+1 cells) `<rate> <burst>` (for CLP=0 cells). `ABR` specifies an ABR connection. There is no format currently defined for ABR.
`<rate>`	Specifies the rate for the QoS class specified. For UBR and CBR, this is the Peak Cell Rate (PCR) in cells per second. For VBR and NRT-VBR, this is the PCR for cells that are CLP=0+1 in cells per second, the Sustainable Cell Rate (SCR) for cells that are CLP=0+1 in cells per second, and the SCR for cells that are CLP=0 in cells per second.
`<burst>`	Specifies the Maximum Burst Size (MBS) in cells for the QoS class specified (VBR and NRT-VBR only). This is the MBS for cells that are CLP=0+1, in cells, and the MBS for cells that are CLP=0, in cells.

Lines beginning with # may be inserted if you wish to include comments or to improve the clarity of the presentation when the file is viewed or printed. These lines are ignored when the file is read. Lines may be continued by escaping the end-of-line with a backslash “\” (do not enter the quote marks).

3.6.1.3 Defining an ELAN

Each ELAN is defined by an address statement whose value denotes the ATM address of the ELAN's LES. For example:

```
engineering.Address: C5.0005.80.ffe100.0000.f21a.01b9.0020480605b2.00
```

NOTE

To configure DLE for an ELAN, use the anycast address in this statement. Be sure to use a distinct anycast address for each ELAN in the network. It must be unique within the first 19 bytes.

In addition, you may instruct a given ELAN to override any of the default values. For example, the `engineering` ELAN could override a `Maximum_Frame_Size` setting of 1516; thus:

```
engineering.Maximum_Frame_Size: 4544
```

If you want to control which clients may or may not join a given ELAN, two additional keys, `Accept` and `Reject`, whose values are comma-separated lists of matching elements, may be used.

NOTE

The `Accept` values must be supplied if you are going to enable ELAN access control.

These values may be:

a MAC address,

```
engineering.Accept: 0020480605b2 , 002048080011 , 0020481020ef
```

an ATM address and equal-length bit mask,

```
engineering.Accept: 47.0005.80.FFE100.0000.0000.0000.002048000000.00\
                    FF.FFFF.FF.FFFFFF.0000.0000.0000.FFFFFF000000.00
```

or an ATM address containing "don't-care" semi-octets denoted by an "X":

```
marketing.Accept: 47.0005.80.FFE100.XXXX.XXXX.XXXX.002048XXXXXX.XX
```

The last two forms of ATM-address matching elements are functionally the same. The latter is shorter but only allows for masks whose semi-octets are all ones or all zeros, while the former allows for arbitrary masks. A prospective-client address is “captured” by an ELAN name if the client’s address matches one of the Accept elements but not one of the Reject elements (if present). Finally, an ELAN may be configured to accept any client that wishes to join by including the following statement:

```
default.Accept:XX.XXXX.XX.XXXXXX.XXXX.XXXX.XXXX.XXXXXXXXXXXX.XX
```

The order in which to apply the Accept and Reject rules is given by a Match.Ordering group.key statement, whose value is a comma-separated list of ELAN names. For example:

```
Match.Ordering: default, engineering, marketing
```

The names of all ELANs that have Accept keys must be included in Match.Ordering.

The LE_CONFIGURE_REQUEST frame contains an ATM address and an optional MAC address or route descriptor (which is always present for ELAN access control requests). The Accept/Reject checking proceeds in two distinct phases: first, for the MAC address or route descriptor, if present, and second, for the ATM address. So, even though a client might be rejected by its MAC address, it can be accepted by its ATM address. Therefore, when configuring the Accept and Reject rules, ensure that you write them either as explicit lists of only MAC addresses or route descriptors, or only as ATM address matches.

3.6.1.4 Defining a Client

Clients need not be defined in the LECS configuration file. Typically, you would define a client for the purpose of overriding one or more of the default configuration parameters for that particular client.

A client is defined by using its ATM address, MAC address, or route descriptor in the group field, and perhaps giving the name of its ELAN as the value of the LAN_Name key. For example:

```
47.0005.80.FFE100.0000.0000.0000.002048222222.22.LAN_Name:engineering
002048ABCDEF.LAN_Name: marketing
012F.LAN_Name: publications
```

Configuration parameter overrides can also be given on a per-client basis. For example, the following statements override the default VCC_TimeOut_Period and Aging_Time configuration parameters for a client whose MAC address is 002048080011 on the engineering ELAN:

```
002048080011.LAN_Name:engineering
002048080011.VCC_TimeOut_Period:1200
002048080011.Aging_Time: 30
```

3.6.1.5 LECS Control Parameters

Specifying values for keys in the `LECS` group provides control over the operation of the LECS process.

NOTE

If you change the values of the LECS control parameters while the LECS process is running, the new values do not take effect until the LECS process is stopped, and then restarted.

When a client contacts the LECS, the connections established are known as Configuration Direct VCCs. To override the default value of the `VCC_TimeOut_Period` key (the number of seconds before an idle Configuration Direct VCC is automatically closed by the LECS), enter a statement similar to the following:

```
LECS.VCC_TimeOut_Period: 1200
```

The LECS periodically checks whether its configuration file has been modified, and, if it has, the file is reread. The length of this period, in seconds, is given by the `Reload_Period` key:

```
LECS.Reload_Period: 600
```

The `Permanent_Circuits` key holds a comma-separated list of `VPI.VCI` pairs denoting the local ends of 0.17 PVCs on which the LECS should listen. For example:

```
LECS.Permanent_Circuits: 0.42, 0.112
```

The LECS can provide the client with a fourteen-bit pattern to permute the MAC-address generation algorithm. This bit pattern is specified with the `MAC_Address_Base` key.

```
LECS.MAC_Address_Base: 38fe
```

3.6.1.6 LECS MPOA Parameters

MPOA requests use similar, and in many cases, the same database keys as LANE requests. However, there are some MPOA-specific keys that can be specified. LEC/MPC parameters can be specified for shortcuts. For example, the following group of parameters indicates that shortcuts should be established for IP flows, but only after a threshold of 10 frames per second is surpassed.

```
.ShortCut_Protocols: IP
.ShortCut_Threshold: 10/1
```

LEC/MPC parameters can also be specified for resolution requests. When a LEC/MPC sends an MPOA Resolution Request, it sets a timer to a Resolution_Initial_Retry_Time. If an MPOA Resolution Reply is not received in that amount of time, a retry may be sent. Each time a retry is sent, the timer is set to the Resolution_Initial_Retry_Time value * a retry multiplier. If the value exceeds the Resolution_Maximum_Retry_Time value, the LEC/MPC assumes the request has failed. A new request may not be sent until the Resolution_Hold_Down_Time has been surpassed.

For example, the following parameters indicate that an initial MPOA Resolution Request should be retried after 5 seconds, backing off to a maximum retry time of 40 seconds, and then the MPOA Resolution Request process is re-initialized after 160 seconds.

```
.Resolution_Initial_Retry_Time: 5
.Resolution_Maximum_Retry_Time: 40
.Resolution_Hold_Down_Time: 160
```

MPOA server (MPS) parameters can also be specified. For example, the following parameters indicate that the MPS must send Keep-Alive messages every 10 seconds. Each of these messages is valid for 35 seconds. The MPS must wait 40 seconds before giving up on a pending resolution request, and should use 1200 seconds as the default holding time in NHRP Resolution Replies.

```
.MPOA_KeepAlive_Time: 10
.MPOA_KeepAlive_Lifetime: 35
.Resolution_GiveUp_Time: 40
.Resolution_Default_Holding_Time: 1200
```

Parameters can also be specified for flow descriptors which determine whether and when to trigger creation of shortcut circuits. The LECS sends the LEC/MPC this set of parameters. These parameters consist of the following elements in the following order: a source/destination specifier, flow establishment thresholds, and a QoS descriptor.

For example, you could specify that telnet traffic to the Class C 202.19.88.0 subnet should be sent on a UBR VCC with a peak rate of 10,000 cells per second, but only after the traffic on that connection exceeds 20 frames per second. If the VCC is idle for more than 10 minutes, then the shortcut should be torn down.

```
.IP_Flows:0.0.0.0/0 202.19.88.0/24 \
             TCP 0 23 \
          20/1 1/600 0/1 \
            0x0 UBR 10000
```

The parameters in the IP flow example are defined as follows:

0.0.0.0/0 The source address / prefix length.

202.19.88.0/24 The destination address / prefix length.

TCP 0 23 The protocol to which this flow applies, and the source and destination ports, respectively.

20/1 1/600 0/1 The set-up threshold, the tear-down threshold, and the set-up threshold for the default forwarding path, respectively.

0x0 UBR 10000 The QoS flag (currently unused and should be set to 0), the QoS class, and the PCR for the specified QoS class, respectively.

3.6.2 Sample LECS Configuration File

CAUTION

Do not attempt to edit an existing functional LECS configuration file without first making a backup copy of the file. Incorrect modification of the configuration file could result in loss of communication between one or more members of a defined ELAN.

NOTE

For a detailed discussion of how to configure an LECS configuration file similar to the one given in this section, please refer to Section 3.6.1.

The sample LECS configuration file shown at the end of this section in Figure 3.19 and Figure 3.20 defines three ELANs:

- default
- engineering
- marketing

The `Match.Ordering` statement specifies the ELAN names in the order that prospective clients will attempt to match. The default configuration parameters are shown with their default values. These values apply to all ELANs in this configuration file, unless overridden for a particular ELAN or client.

ELAN `default` is configured to accept any client that wishes to join. The ATM address of the `default` LES is listed in the `default.Address` statement. If DLE is being configured for ELAN `default`, then this address must be the anycast address that allows the clients to reach any of the DLE peer servers for ELAN `default`.

NOTE

Be sure to choose a distinct anycast address for each ELAN in the network. It must be unique within the first 19 bytes.

ELAN `engineering` has overridden the default `Maximum_Frame_Size` with a new size of 4544 bytes. Consequently, this frame size applies only to traffic on the `engineering` ELAN. The `default` and `marketing` ELANs continue to use the default frame size of 1516 bytes.

Two LECs, whose MAC addresses are `002048080011` and `0020481020ef`, are identified as acceptable clients for the `engineering` and `marketing` ELANs.

```
#
# The search ordering of elan names
#
Match.Ordering: default, engineering, marketing
#
# the default configuration parameters
#
.Control_TimeOut: 120
.Maximum_Unknown_Frame_Count: 1
.Maximum_Unknown_Frame_Time: 1
.VCC_TimeOut_Period: 1200
.Maximum_Retry_Count: 1
.Aging_Time:  300
.Forward_Delay_Time: 15
.Expected_LE_ARP_Response_Time: 1
.Flush_TimeOut: 4
.Path_Switching_Delay: 6
.Multicast_Send_VCC_Type: Best Effort
.Connection_Complete_Timer: 4
.LAN_Type: Ethernet/IEEE 802.3
.Maximum_Frame_Size: 1516
.ShortCut_Protocols:IP
.ShortCut_Threshold:10/1
.Resolution_Initial_Retry_Time:5
.Resolution_Maximum_Retry_Time:40
.Resolution_Hold_Down_Time:160
.MPOA_KeepAlive_Time:10
.MPOA_KeepAlive_Lifetime:35
.Resolution_GiveUp_Time:40
.Resolution_Default_Holding_Time:1200
.IP_Flows:   0.0.0.0/0 202.19.88.0/24 \
             TCP 0 23 \
             20/1 1/600 0/1 \
             0x0 UBR 10000
#
# Parameters for the default elan
#
default.Address: C5.0005.80.ffe100.0000.f21a.21b8.0097036324b2.25
default.Accept: XX.XXXX.XX.XXXXXX.XXXX.XXXX.XXXX.XXXXXXXXXXXX.XX
```

Figure 3.19 - Sample LECS Configuration File (Part One of Two)

```
#
# Parameters for elan: engineering
#
engineering.Address:C5.0005.80.ffe100.0000.f21a.01b9.0020480605b2.11
engineering.Accept: 002048080011 , 0020481020ef
engineering.Maximum_Frame_Size: 4544
#
# Parameters for elan: marketing
#
marketing.Address: C5.0005.80.ffe100.0000.f21a.01b9.0020480605b2.21
marketing.Accept:  002048080011 , 0020481020ef
#
```

Figure 3.20 - Sample LECS Configuration File (Part Two of Two)

3.6.3 Starting the LAN Emulation Services

LAN Emulation services include the LECS and the LES/BUS. Once the LECS configuration database file has been configured, these services must be started so that they are available for LECs to attempt to use. Using *ForeThought* 5.0 or greater, the LES/BUS services must run in the same device.

NOTE

LANE services running on a switch are LANE 2.0 compliant, but LANE clients running on a switch are LANE 1.0 compliant.

3.6.3.1 Starting the LECS

Once an LECS configuration file has been configured, you need to retrieve the LECS configuration database file that you built elsewhere and put it on the switch that is going to run the LECS.

1. Use the following AMI command to retrieve the file:

```
configuration lane lecs get <host>:<remotefile> [<localfile>]
```

For example, you would enter something similar to the following:

```
configuration lane lecs get 198.29.22.46:lecs.cfg
```

2. After you have retrieved the LECS configuration database file, use the following AMI command to start the LECS service on the switch:

```
conf lane lecs new <LECS Selector byte (HEX)> [-db <LECS database
     file>]\[-default <LES atm address>] [<LECS-wka> | none]
```

For example, to start the LECS service using the `-db` option and using the ATM Forum's well-known address you would enter something similar to the following:

```
configuration lane lecs new 0x0c -db lecs.cfg
```

NOTE

If you want to use an address other than the ATM Forum's address, you would enter that address at the end of the command. If you want to disable the well-known address so that the LECS can only be contacted by using the switch's actual address (with selector byte) type **none** at the end of the command.

3. Use the following AMI command to verify that the LECS has been started and is running:

```
configuration lane lecs show
```

```
Index  AdminStatus  OperStatus   Selector  WKA         Database
    1  up           up               0x0c  atm-forum   lecs.cfg
```

The `OperStatus` field shows `up`, meaning that the LECS is running. Now you must start the DLE peer servers as described in the next section.

NOTE

If you used an address other than the ATM Forum's address, the `WKA` field would show `other` and that address would be displayed below the entry. If you disabled the well-known address the `WKA` field would show `none`.

NOTE

If you used the `-default` option, then that LES address would be displayed below this entry in a field titled `Default LES`.

3.6.3.2 Starting the DLE LES/BUS Peer Servers

The LES and BUS services must be started for the ELAN. This example assumes you are using DLE; therefore, you must enter a distinct anycast address (that is unique within the first 19 bytes) for the LECs to use to contact the LES, and the address of each of the DLE peer servers so that this server can communicate with its peers.

1. To start the services, use the following AMI command on the switch that is going to run one of the DLE peer servers:

```
conf lane les new <LES Selector Byte (HEX)> <LES name>\
           [-bus <BUS Selector Byte (HEX)>]\
           [-type (ethernet | token-ring)] \
     [-mtu (1516 | 1580 | 4544 | 9234 | 18190)] \
          [-secure wka | <LECS ATM Address>] \
          [-registertlvs (enable | disable)] \
                 [-elanid <ELAN ID>] \
               [-segid <TR Segment ID>] \
             [-fwdarp (enable | disable)] \
         [-anycast <LES Anycast ATM Address>]\
                [-peers <atm-addr> ...]
```

For example, you would enter something similar to the following:

```
       conf lane les new 90 engineering -anycast
c5.0005.80.ffe100.0000.f21a.3596.0020481a3596.f0 -peers
   47.0005.80.ffe100.0000.f21a.10bb.0020481a10bb.90
   47.0005.80.ffe100.0000.f21a.3552.0020481a3552.10
   47.0005.80.ffe100.0000.f21a.3218.0020481a3218.44
```

NOTE

This command creates a co-located LES and BUS using a single AMI command. You cannot create a BUS separately using *ForeThought* 5.0 or greater. The **conf lane bus** commands are only useful in providing backwards compatibility with switches that are running earlier versions of *ForeThought* software.

NOTE

If no ELAN type is entered, the switch assumes Ethernet and uses 1516 as the MTU size. If Token Ring is used as the type, but no MTU size is entered, 4544 is used as the size.

NOTE

You must enter the address of each of the DLE peer servers when you are starting DLE; e.g., if you want four peers, then all four must be configured with the addresses of the other three peers, as well as their own LES address, at the time that each LES/BUS is started. If you want to add new peers to the list, you can use the **`conf lane les peeradd`** command.

2. Use the following AMI command to verify that the LES and the BUS have been started and are running.

```
configuration lane les show
```

```
      Admin Oper  Les                  ELAN                               ELAN
Index State State Sel   Type       MTU   Name        SECURE  FwdTLVs FwdARPs ID
1     up    up    0x90 ethernet   1516  engineering disable enable  disable 100
       LES : 0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.10
       BUS : 0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.10 (Co-Located)
           : c5000580ffe1000000f21c10bb0020323c10bb10 (ANYCAST)
       LECS   :0xc5.0079.00.000000.0000.0000.0000.00a03e000001.00
       PEER   :0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.10
       PEER   :0x47.0005.80.ffe100.0000.f21c.10bb.0020481a42bb.10
       PEER   :0x47.0005.80.ffe100.0000.f21c.10bb.0020482a55a0.70
```

The `OperStatus` field shows `up`, meaning that the LES and the BUS are running.

3. Configure the DLE peer servers by opening an AMI session on one of the other switches that will run a DLE peer server and enter something similar to the following:

```
conf lane les new 10 engineering -anycast
c5.0005.80.ffe100.0000.f21a.3596.0020481a3596.f0 -peers
47.0005.80.ffe100.0000.f21a.3552.0020481a3552.10
47.0005.80.ffe100.0000.f21a.10bb.0020481a10bb.90
47.0005.80.ffe100.0000.f21a.3218.0020481a3218.44
```

4. Open an AMI session on the third switch that will run a DLE peer server and enter something similar to the following:

```
       conf lane les new 44 engineering -anycast
c5.0005.80.ffe100.0000.f21a.3596.0020481a3596.f0 -peers
    47.0005.80.ffe100.0000.f21a.3218.0020481a3218.44
    47.0005.80.ffe100.0000.f21a.10bb.0020481a10bb.90
    47.0005.80.ffe100.0000.f21a.3552.0020481a3552.10
```

   Once all of the peers have been created, use the **conf lane les show advanced** command to verify that the peers have established point-to-point and point-to-multipoint connections to each other.

3.6.4 Starting the LEC(s) and Joining an ELAN

Now that the ELAN services have been started, you can have LECs join the ELAN that you have created.

NOTE

The switch software only allows you to create an instance of a LEC on a switch. To create an instance of a LEC on a host, you must use the *ForeRunner* VLAN Manager or use a *ForeRunner* host adapter. Please refer to the respective User's Manual for instructions.

NOTE

A LEC created on the switch cannot join a Token Ring ELAN. It can only join an Ethernet ELAN.

1. To start a LEC that will attempt to join the ELAN, use the following AMI command on the switch that is going to be a LEC:

```
conf lane lec new <LEC Selector byte (HEX)> <ELAN name>
        [-ip <IP Address> [-mask <IP netmask>]]
                 [(wellknown | manual)]
manual mode options: [-lecs <LECS address>] or [-les <LES address>]
```

NOTE

The recommended (and default) method for starting a LEC is to use the `wellknown` mode, meaning that the LEC will attempt to contact the LECS on the "well-known" address as defined by the ATM Forum's LAN Emulation standards (`47.0079.00.000000.0000.0000.0000.00A03E000001.00`).

For example, to start a LEC that attempts to join the ELAN called engineering, enter the following:

```
configuration lane lec new 2 engineering -ip 192.168.61.25 -mask
                        255.255.255.0
```

NOTE

If you want to use the `manual` mode, you must enter either a LECS address other than the well-known address or you must enter a LES address. If you enter a LES address, this means that the LEC bypasses the LECS and directly contacts the specified LES.

NOTE

In order to use DLE, the LES address to be used must be the anycast address of the DLE peer server, not the server NSAP address.

2. Verify that the LEC has joined the ELAN by using the following AMI command:

`configuration lane lec show`

```
       Admin   Oper
Index  Status  Status  Sel   Mode         MACaddress    IfName    ELAN
    1  up      up      0x02  wellknown    00204815096b  el0       engineering
       LECS:0x47.0079.00.000000.0000.0000.0000.00a03e000001.00
       LES :c000580ffe1000000f21a35960020481a3596f0
```

The `OperStatus` field shows `up`, meaning that the LEC has successfully joined the ELAN.

NOTE

If the `OperStatus` field shows `joining`, this means that the LEC is still registering with the ELAN. Wait a few seconds and check it again. When it has finished, the `OperStatus` field displays `up`.

You can also use the command **`conf lane lec show advanced`** to display more information about the LEC.

3. After the first LEC has joined the ELAN, you can perform the same steps in this section on another switch to allow a LEC to run on that switch. You can also use the VLAN Manager or the host software to add more LECs to this ELAN. Once all the LECs have joined, the ELAN is complete.

3.7 Upgrading an ELAN to Use DLE

This section describes how to upgrade your ELAN if you had previously configured LANE, you want to upgrade some or all of the clients from *ForeThought* 4.1.x to *ForeThought* 6.x, and you want to upgrade all the equipment that is running services to *ForeThought* 6.x using DLE.

NOTE

If you used the VLAN Manager to create your ELAN, you should use it to upgrade the ELAN. Refer to the *ForeView VLAN Manager User's Manual* for the appropriate steps.

The following basic steps are involved in upgrading your ELAN to use DLE. Each of these steps is described in detail in the following sections. It is recommended that you read the entire section before attempting to upgrade to DLE. These steps describe the commands for a switch. Refer to the corresponding User's Manual for the appropriate steps for commands for hosts, *PowerHubs*, ES-3810s, or the VLAN Manager.

1. Edit the LECS.CFG file.
2. Delete the old LES and BUS.
3. Upgrade the switches that are running services.
4. Create the DLE peer servers.
5. Transfer the updated LECS.CFG file.
6. Restart the LECS.
7. Recreate the LECs.
8. Create the last DLE peer.
9. Add the last DLE peer to each list of peers.
10. Change the last failover ELAN information in the LECS.CFG file.
11. Transfer the final LECS.CFG file.
12. Restart the LECS.

3.7.1 Edit the LECS.CFG File

NOTE

Before you edit the LECS.CFG file, you may wish to back it up to a host using the **oper flash put** command. If you are using TFTP as the transfer protocol (this is the default), the remote host to which the FLASH file will be sent must be running the TFTP server code and must have on an empty file in the `/tftpboot` directory on the remote host to receive the FLASH file. See Chapter 3 in the *ATM Management Interface (AMI) Manual* for more information. If you are using FTP as the transfer protocol, you only need to enter the command shown below to perform the transfer.

Transfer the LECS.CFG file that is currently being used by the LECS to a workstation (other than the one to which you backed up the file) so you can edit the file.

```
myswitch::operation> flash put lecs.cfg 169.144.85.195:/tftpboot/lecs.cfg
Transferred 2323 bytes of fs:/lecs.cfg
```

The information about the LES addresses in your old LECS.CFG file may look something like this before editing:

```
Match.Ordering:         Mktg|0, Mktg|1

Mktg|0.Address:   47000580ffe1000000f21a24f90020481a24f900
Mktg|0.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Mktg|1.Address:   47000580ffe1000000f21a390e0020481a390e00
Mktg|1.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

On the workstation that has the LECS.CFG file, use a text editor to make the following changes:

1. Add a new ELAN to the beginning of the Match.Ordering list that is named like the old failover ELANs, but without the `|n`. In this example, there are currently two failover ELANS named Mktg|0 and Mktg|1. Therefore, add an ELAN named Mktg.

2. Give the anycast address of the DLE peer servers to the new ELAN. Also, replace the LES address of the old ELANs with this anycast address, except for the last of the |n failover ELANs (in this case, Mktg|1). Leave Mktg|1 with old LES address for now so the LECs can use it until they are all changed over. (It will be changed at the end of the process.)
3. If there were any non-colocated BUSs (none shown in this example), delete any lines that refer to them, since each BUS will co-located with a LES after you upgrade to *ForeThought* 6.x.

After following the steps above, the information about the LES addresses in your LECS.CFG file would look like this:

```
Match.Ordering:         Mktg, Mktg|0, Mktg|1

Mktg.Address:     c5000580ffe1000000f21c126b0020481c126b66
Mktg.Accept:      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Mktg|0.Address:   c5000580ffe1000000f21c126b0020481c126b66
Mktg|0.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Mktg|1.Address:   47000580ffe1000000f21a390e0020481a390e00
Mktg|1.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

3.7.2 Delete the LES and BUS

Display the LES information so that you can find the index number of the LES.

```
myswitch::configuration lane les> show
 Index AdminStatus OperStatus Selector  Type       MTU   ELAN
     1 up          up         0x00      ethernet   1516  Mktg|0
        LES : 0x47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
        BUS : 0x47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00 (Co-Located)
```

Administer the LES and BUS down for Mktg|0 using the following AMI command:

```
myswitch::configuration lane les> admin 1 down
```

Delete the co-located LES and BUS.

```
myswitch::configuration lane les> delete 1
```

Verify the LES has been deleted.

```
myswitch::configuration lane les> show
No LES information is available.
```

NOTE

The LES and BUS in this example were co-located. If yours are not co-located, you need to administer the BUS down using **conf lane bus admin** <*index*> **down** and to delete the BUS using **conf lane bus delete** <*index*>. The BUS will be automatically created as a co-located BUS when you use the **conf lane les new** command.

Do the same for each old LES and BUS, except for the last LES and BUS (in this case, Mktg|1). All clients will still be using the last LES/BUS temporarily until the DLE peer servers have been established. The last LES and BUS will be changed over at the very end of the process.

3.7.3 Upgrade the Switches Running Services

Upgrade each of the switches that is going to be running services to *ForeThought* 6.x using the following AMI command:

oper upgrade *<remotehost>*:*<full path to remotefile>*

Do not upgrade the switch that is running the last LES and BUS (in this case, Mktg|1). The LECs will still be using the last LES/BUS temporarily until the DLE peer servers have been established.

3.7.4 Create the DLE Peer Servers

NOTE

You must enter the address of each of the DLE peer servers when you are starting DLE; e.g., if you want four peers, then all four must be configured with the addresses of the other three peers, as well as their own LES address, at the time that each LES/BUS is started. If you want to add new peers to the list, you can use the **conf lane les peeradd** command.

Create a DLE peer server (LES/BUS pair) with the new ELAN name (in this case, Mktg) on each switch that is to become a DLE peer server.

```
myswitch::configuration lane les> new 0x00 Mktg -anycast
c5.0005.80.ffe100.0000.f21c.126b.0020481c126b.66 -peers
47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90
```

Use the **show** command to verify the information that you entered:

```
myswitch::configuration lane les> show
       Admin Oper  Les                   ELAN                            ELAN
 Index State State Sel  Type       MTU   Name     SECURE  FwdTLVs FwdARPs ID
 1     up    up    0x00 ethernet   1516  Mktg     disable enable  disable 100
        LES : 0x47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
            :   c5.0005.80.ffe100.0000.f21c.126b.0020481c126b.66 (ANYCAST)
        BUS : 0x47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00 (Co-Located)
        PEER: 0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90
```

Create each of the other peer servers using the same anycast address and giving them the peer address(es) of each peer in the ELAN. (In this example, there is only one peer.)

```
myswitch::configuration lane les> new 0x90 Mktg -anycast
c5.0005.80.ffe100.0000.f21c.126b.0020481c126b.66 -peers
47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90
47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
```

Use the **show** command to verify the information that you entered:

```
myswitch::configuration lane les> show
       Admin Oper  Les                   ELAN                            ELAN
 Index State State Sel  Type       MTU   Name     SECURE  FwdTLVs FwdARPs ID
 1     up    up    0x90 ethernet   1516  Mktg     disable enable  disable 100
        LES    :0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90
        BUS    :0x47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90 (Co-Located)
               :c5000580ffe1000000f21c126b0020481c126b66 (ANYCAST)
        PEER   :0x47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
```

3.7.5 Transfer the Updated LECS.CFG File

Transfer the updated LECS.CFG file back to the switch that is running the LECS.

```
myswitch::operation flash> get 169.144.85.195:/tftpboot/lecs.cfg lecs.cfg
```

3.7.6 Restart the LECS

Administer the LECS down and back up again. This forces any active clients to re-establish their connection with the LECS and forces the LECS to read and use the new LECS.CFG file.

```
        myswitch::configuration lane lecs> admin 1 down

        myswitch::configuration lane lecs> admin 1 up
```

Use the **show** command and look at the `OperStatus` field to verify that the LECS has come up again.

```
        myswitch::configuration lane lecs> show
         Index  AdminStatus  OperStatus   Selector  WKA          Database
             1  up           up               0x00  atm-forum    lecs.cfg
```

3.7.7 Recreate the LECs

If you are going to upgrade any or all of the clients to *ForeThought* 6.x, you should do so now. Use the following AMI command to upgrade any of the clients that are running on switches:

oper upgrade *<remotehost>*:*<full path to remotefile>*

NOTE

The rest of these instructions apply whether you upgraded the LEC on the switch to *ForeThought* 6.x or not. For commands to recreate LECs that are on platforms other than a switch, refer to the appropriate User’s Manual.

Use the following command to get the interface name from the IfName field as follows:

```
myswitch::configuration lane lec> show advanced
        Admin   Oper
 Index  Status  Status   Sel   Mode       MACaddress    IfName    ELAN
     1  up      up       0x11  wellknown  0a20481a2c78  el17      Mktg|0
        LECS:0x47.0079.00.000000.0000.0000.0000.00a03e000001.00
        LES :c5000580ffe1000000f21c126b0020481c126b66
        BUS : 47000580ffe1000000f21c126b0020481c126b66
        LEC ID : 13057                          Discovered ELAN name : Mktg|0
        Configure Direct VCC : 0.323            Maximum Frame Size : 1516
        Control Direct VCC : 0.667              Control Distribute VCC : 0.203
        Multicast Send VCC : 0.674              Multicast Forward VCC : 0.205
        Last Error :
```

Configure that LEC down.

```
myswitch::configuration> ip admin el17 down
```

Delete all instances of the LEC (including anything that had been <*ELAN name*>|n).

```
myswitch::configuration lane lec> delete 1

myswitch::configuration lane lec> delete 2
```

Recreate the LEC. (You now only need one instance per LEC and you will not specify |n in the name anymore.)

```
myswitch::configuration lane lec> new 0x11 Mktg -ip 192.168.61.25 -mask 255.255.255.0
```

Use the following command to verify that the LEC has joined the ELAN by looking at the `OperStatus` field as follows:

```
myswitch::configuration> lane lec show advanced
        Admin   Oper
 Index  Status  Status   Sel   Mode       MACaddress    IfName    ELAN
     1  up      up       0x11  wellknown  0a20481a2c78  el17      Mktg
        LECS:0x47.0079.00.000000.0000.0000.0000.00a03e000001.00
        LES :c5000580ffe1000000f21c126b0020481c126b66
        BUS : 47000580ffe1000000f21c126b0020481c126b66
        LEC ID : 13057                          Discovered ELAN name : Mktg
        Configure Direct VCC : 0.323            Maximum Frame Size : 1516
        Control Direct VCC : 0.667              Control Distribute VCC : 0.203
        Multicast Send VCC : 0.674              Multicast Forward VCC : 0.205
        Last Error :
```

Repeat all of the steps in this section for each LEC. Refer to the corresponding User's Manual for the appropriate steps for re-creating LECs that are running on hosts, *PowerHub*s, or ES-3810s. At this time, you may also add any new clients to the ELAN.

3.7.8 Create the Last DLE Peer

After all of the LECs have been changed over and any new LECs have been added, go back and change the remaining failover LES to a DLE peer. Create the DLE peer as follows:

```
myswitch::configuration lane les> new 0x02 Mktg -anycast
c5.0005.80.ffe100.0000.f21c.126b.0020481c126b.66 -peers
47.0005.80.ffe100.0000.f21a.24f9.0020321a26e5.02
47.0005.80.ffe100.0000.f21a.24f9.0020481a24f9.00
47.0005.80.ffe100.0000.f21c.10bb.0020481c10bb.90
```

3.7.9 Add the Last DLE Peer to Each Peer List

Now add the new peer to the list of each of the existing peers using the following command on each of those peers:

```
myswitch::configuration lane les> new -peeradd 1
47.0005.80.ffe100.0000.f21a.24f9.0020321a26e5.02
```

3.7.10 Update the LECS.CFG File

After you add the last DLE peer, go back and delete the old remaining failover ELAN information from the LECS.CFG file. Transfer the file to a workstation for editing.

```
myswitch::operation> flash put lecs.cfg 169.144.85.195:/tftpboot/lecs.cfg
Transferred 2323 bytes of fs:/lecs.cfg
```

On that workstation, use a text editor to delete all lines referring to any of the `|n` failover ELANs (in this case, Mktg|0 and Mktg|1). If the LES address portion of your LECS.CFG file looked like this before editing:

```
Match.Ordering:         Mktg, Mktg|0, Mktg|1

Mktg.Address:     c5000580ffe1000000f21c126b0020481c126b66
Mktg.Accept:      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Mktg|0.Address:   c5000580ffe1000000f21c126b0020481c126b66
Mktg|0.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

Mktg|1.Address:   47000580ffe1000000f21a390e0020481a390e00
Mktg|1.Accept:    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

It will look like this after editing:

```
Match.Ordering:         Mktg

Mktg.Address:     c5000580ffe1000000f21c126b0020481c126b66
Mktg.Accept:      XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

3.7.11 Transfer the Final LECS.CFG File

Transfer the final LECS.CFG file back to the switch that is running the LECS.

```
myswitch::operation flash> get 169.144.85.195:/tftpboot/lecs.cfg lecs.cfg
```

3.7.12 Restart the LECS

Administer the LECS down and back up again. This forces the active LECs to re-establish their connection with the LECS and forces the LECS to read and use the new LECS.CFG file.

```
myswitch::configuration lane lecs> admin 1 down

myswitch::configuration lane lecs> admin 1 up
```

Use the **show** command and look at the `OperStatus` field to verify that the LECS has come up again:

```
myswitch::configuration lane lecs> show
 Index  AdminStatus  OperStatus   Selector  WKA         Database
     1  up           up               0x00  atm-forum   lecs.cfg
```

The transfer of your ELAN to DLE services is now complete.

3.8 Upgrading an ELAN without Using DLE

This section describes how to upgrade your ELAN if you had previously configured LANE, you want to leave the clients running *ForeThought* 4.1.x, and you want to upgrade all the equipment that is running services to *ForeThought* 6.x **without** using DLE.

The following basic steps are involved in upgrading your ELAN *ForeThought* 6.x **without** using DLE. Each of these steps is described in detail in the following sections. It is recommended that you read the entire section before attempting the upgrade.

1. If the LES and BUS are not co-located, they must be deleted and re-created as co-located because *ForeThought* 6.x on the switch only supports co-located services. If the LES and BUS are already co-located, skip to step 2.
2. Upgrade the switches that are running services.
3. Re-create the LES and BUS if they were not co-located. If the LES and BUS were already co-located, skip to step 4.
4. Administer up the switches that are running services.

NOTE

If you used the VLAN Manager to create your ELAN, you should use it to upgrade the ELAN. Refer to the *ForeView VLAN Manager User's Manual* for the appropriate steps.

NOTE

These steps describe commands for services running on a switch. Refer to the corresponding User's Manual for the appropriate steps for commands for any services that are running on hosts, *PowerHubs*, or ES-3810s.

3.8.1 Deleting the Non Co-located Services

There are several steps involved in changing the non co-located services.

3.8.1.1 Administer Down the Services

Administer down the LECS.

```
myswitch::configuration lane> lecs admin <LECS index> down
```

Administer down the LES and BUS.

```
myswitch::configuration lane> les admin <LES index> down

myswitch::configuration lane> bus admin <BUS index> down
```

3.8.1.2 Delete the Non Co-located LES and BUS

Delete the LES and BUS.

```
myswitch::configuration lane> les delete <LES index>

myswitch::configuration lane> bus delete <BUS index>
```

3.8.1.2.1 Edit the LECS.CFG File

NOTE

Before you edit the LECS.CFG file, you may wish to back it up to a host using the **`oper flash put`** command. If you are using TFTP as the transfer protocol (this is the default), the remote host to which the FLASH file will be sent must be running the TFTP server code and must have an empty file in the `/tftpboot` directory on the remote host to receive the FLASH file. See Chapter 3 in the *ATM Management Interface (AMI) Manual* for more information. If you are using FTP as the transfer protocol, you only need to enter the command shown below to perform the transfer.

Transfer the LECS.CFG file that is currently being used by the LECS to a workstation (other than the one to which you backed up the file) so you can edit the file.

```
myswitch::operation> flash put lecs.cfg 169.144.85.195:/tftpboot/lecs.cfg
Transferred 2323 bytes of fs:/lecs.cfg
```

On the workstation that has the LECS.CFG file, use a text editor to delete any lines that refer to any non-colocated BUSs.

After you have made the changes, transfer the LECS.CFG file back to the LECS.

```
myswitch::operation flash> get 169.144.85.195:/tftpboot/lecs.cfg lecs.cfg
```

3.8.2 Upgrade the Switches Running Services

Upgrade the switch(es) running each of the services to *ForeThought* 6.x using the following AMI command:

```
oper upgrade <remotehost>:<full path to remotefile>
```

3.8.3 Recreate the LES and BUS Together

If your LES and BUS are already co-located, skip to step 4. If your LES and BUS were not co-located, re-create each LES and BUS using the following single AMI command:

```
myswitch::configuration lane les> new <LES Selector Byte (HEX)> <LES name> \
        [-bus <BUS Selector Byte (HEX)>] \
        [-type (ethernet | token-ring)] \
        [-mtu (1516 | 1580 | 4544 | 9234 | 18190)] \
        [-secure wka | <LECS ATM Address>] \
        [-registertlvs (enable | disable)] \
        [-elanid <ELAN ID>] \
        [-segid <TR Segment ID>] \
        [-fwdarp (enable | disable)] \
        [-anycast <LES Anycast ATM Address>] \
        [-peers <atm-addr> ...]
```

You need to specify the LES selector byte (the BUS uses the same selector byte by default) and you need to give the same LES name that you used before. You may optionally specify any of the parameters, except for the anycast address and the peer addresses since you are not using DLE.

3.8.4 Administer the Services Up

Administer up the LES/BUS pair and the LECS on the switch(es) running each of the services.

```
        myswitch::configuration lane> les admin <LES index> up

        myswitch::configuration lane> lecs admin <LECS index> up
```

The upgrade of your ELAN is now complete.

CHAPTER 4 MPOA

This chapter provides an overview of LAN Emulation (LANE) and Multi-Protocol Over ATM (MPOA) as implemented in FORE Systems' *ForeThought* software.

4.1 Overview of LANE/MPOA

FORE System's *ForeThought* software is compliant with the ATM Forum's *LAN Emulation Over ATM Version 1.0* specification. LANE allows higher level protocols and LAN applications to interoperate, without modifications, with an ATM network.

The LANE components, running on the ATM network, interact to emulate an Ethernet or Token Ring LAN. This emulated Ethernet or Token Ring LAN is called an *emulated LAN (ELAN)*. The ELAN components resolve MAC addresses to ATM addresses, replace the connectionless operation of legacy LANs with point-to-point connections, and provide broadcast and multicast services. The ELAN consists of a LANE / MPOA Client (LEC / MPC) running on each host in the ELAN, and the following LANE Services:

- the LAN Emulation Server (LES)
- the Broadcast and Unknown Server (BUS)
- the LAN Emulation Configuration Server (LECS)

The LANE services may operate on a FORE Systems ATM switch, *PowerHub* 7000, *PowerHub* 8000, ASN-9000, or Solaris workstation. *ForeThought* also provides support for Distributed LAN Emulation (DLE) which provides load-sharing and improved fault-tolerance within an ELAN.

4.2 LANE Primer

LANE is the foundation on which MPOA is built. Therefore, before presenting an explanation of MPOA, an understanding of LANE components and their operation in an ELAN is needed.

4.2.1 LANE Components

An ELAN includes the following components:

LANE/MPOA Client (LEC/MPC) The LEC/MPC can wear two different "hats." When wearing its LEC "hat," it simply communicates with other ELAN components (the LES and BUS) to resolve MAC addresses into ATM addresses. When it puts on its MPC "hat," the additional function of the LEC/MPC in an MPOA-aware network is to source and sink internetwork shortcuts.

LAN Emulation Configuration Server (LECS) Runs on a Solaris workstation or a FORE Systems ATM switch. Maintains information about all ELANs within the administrative domain. When the LEC/MPC successfully communicates with the LECS, the LECS provides a list of ELANs which the LEC/MPC can join. The LECS may be configured with various MPOA parameters. LEC/MPCs that connect to LANE/MPOA services through an MPOA-aware LECS are configured with these centrally-supplied MPOA parameters. LEC/MPCs that connect through an LECS that does not contain MPOA parameters still perform flow analysis and attempt inter-ELAN shortcuts according to their user-editable or factory-default settings.

LAN Emulation Server (LES) Runs on a *PowerHub* 7000, a *PowerHub* 8000, an ASN-9000, a FORE ATM switch, or a Solaris workstation. Maintains information about the LEC/MPCs within a single ELAN and performs address resolution. The LES can be configured to support or disable MPOA operation in an ELAN. The LES accepts MPOA parameters from registering LEC/MPCs and MPSs, and also distributes MPOA parameters to LEC/MPCs in response to queries. (This is the mechanism used by LEC/MPCs to determine whether routers in the ELAN are MPOA-aware).

Broadcast and Unknown Server (BUS) Runs on a *PowerHub* 7000, a *PowerHub* 8000, an ASN-9000, a FORE ATM switch, or a Solaris workstation. Provides services within a single ELAN allowing broadcasts, multicasts, and unknown unicasts. The BUS is MPOA-ignorant.

4.2.2 An Example LANE Configuration

Figure 4.1 shows an example configuration of a single ELAN in a FORE network. The ELAN includes:

- PC Workstations, each running a LEC/MPC. Each has a *ForeRunner* ATM adapter, the *ForeRunner* driver for the adapter, and one or more *ForeRunner* ELAN drivers installed.
- Two *ForeRunner ASX*-200BX switches running LESs, BUSs, and LECs. Each switch is also running an LECS. The LES/BUS pairs are configured as *peers* under Distributed LAN Emulation. The peer configuration allows the LECs associated with a particular LES/BUS automatically to reconnect to the remaining functional peer if their “home” LES/BUS fails.
- A *PowerHub* 7000 running a LEC/MPC, and providing access to non-ATM networks.

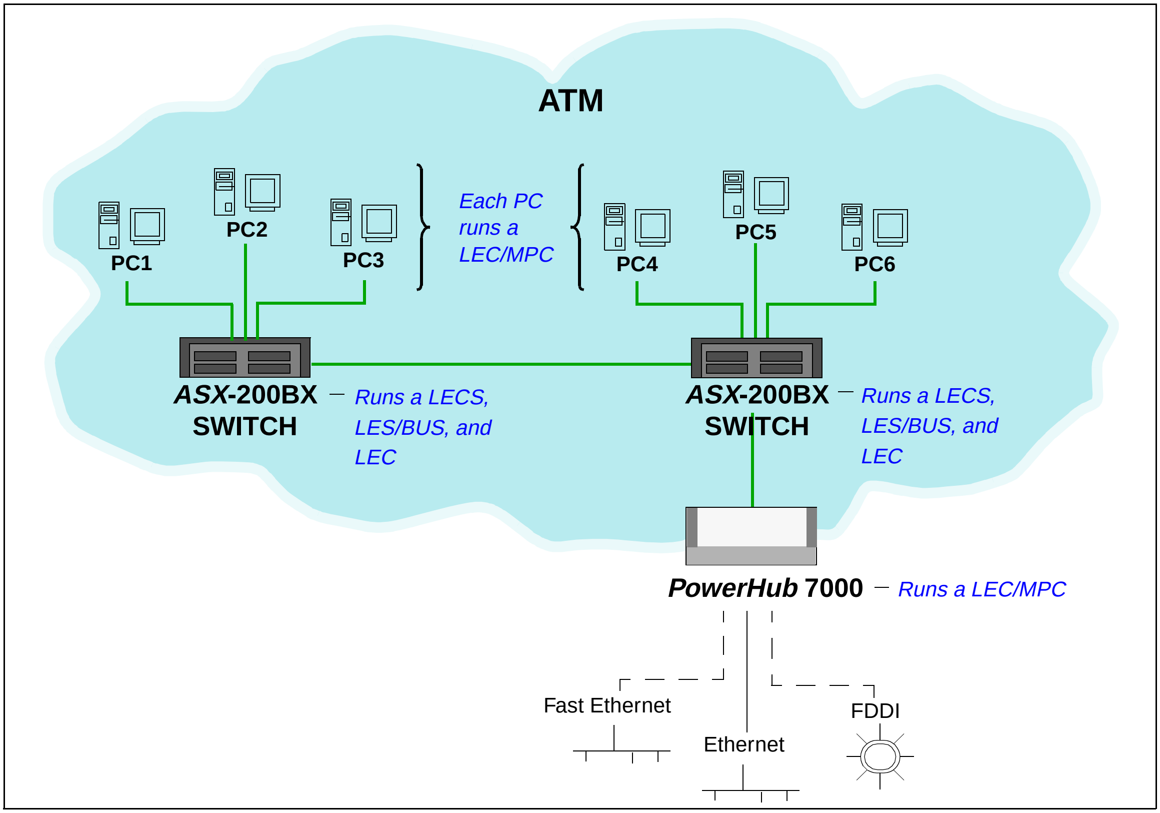

Figure 4.1 - An Example of an ELAN

4.2.2.1 The Initialization Process

Each LEC / MPC goes through the following process when it starts up:

1. The LEC / MPC obtains its own ATM address via address registration. Optionally, you can manually specify the ATM address.
2. The LEC / MPC establishes a connection to a LECS using an address obtained via ILMI, a well-known address, or PVC (0,17). Optionally, you can manually specify the LECS address.

3. The LEC / MPC requests the information needed to join a specified ELAN or the default ELAN. The LECS has information about available ELANs, what ELANs each LEC / MPC can join, and which ELAN the LEC / MPC should attempt to join first.

   If a LECS is not available, or if you choose not to use it, you can manually specify the information required to join a specific ELAN.

4. The LEC / MPC contacts the LES associated with the ELAN it wants to join and registers its MAC-ATM address pair. It also contacts the BUS associated with the ELAN. At this point, the LEC / MPC and the LES have the information required to allow this host to communicate with other hosts on the ELAN as if it were an Ethernet (or Token Ring) network. Refer to the following section for a description of how the LEC / MPC connects to other hosts on the ELAN.

4.2.2.2 The Connection Process

To send packets to another host on the ELAN:

1. The LEC / MPC calls the LES to map the MAC destination address into an ATM address. (The LES maintains a mapping table of the address of all LEC / MPCs on the ELAN.)
2. If the LES finds an entry in its table for the destination MAC address, it returns the destination ATM address to the LEC / MPC.
3. The LEC / MPC then opens up a point-to-point ATM connection to the destination host to send the packet.

4.2.2.3 Multicast and Broadcast Packets

The LEC / MPC sends outgoing multicast and broadcast packets to the BUS which uses a point-to-multipoint connection to send the packets to multiple ATM addresses in the ELAN.

4.2.2.4 Accessing Fast Ethernet and FDDI Networks

Note that the diagram in Figure 4.1 shows dotted lines from the *PowerHub* 7000 to the Fast Ethernet and FDDI networks. This shows how the *PowerHub* can provide access to non-ATM networks.

4.2.2.5 Multiple ELANs

It is possible to set up more than one ELAN in a FORE network. For each new ELAN, you must configure another LES/BUS instance for that LAN. On the access devices, bridge groups must be used to associate physical ports with ELANs on the ATM side. An end station in the ELAN with a *ForeRunner* adapter can connect to up to 16 ELANs simultaneously.

4.2.2.6 Distributed LAN Emulation

To provide greater resilience, support larger ELANs, and support separated clusters of users in an ELAN, *ForeThought* software provides Distributed LAN Emulation (DLE). DLE allows the LES/BUS functions to be distributed among multiple interconnected LES/BUS instances called *peers*. In the example ELAN shown in Figure 4.1, the two LES/BUS pairs running in the switches function as peers in the same ELAN. The LEC/MPCs are distributed such that they are not all connected to the same server. With this arrangement, should one of the peer servers fail, the clients connected to the remaining server continue to maintain connectivity; while the clients that were connected to the failed server automatically re-establish connectivity to the ELAN within 60 seconds.

4.2.2.7 Automatic ELAN Selection

To simplify configuration of the ELAN, a host is allowed to join an ELAN without specifying an ELAN name. If the LECS has been configured to provide the required information, and you do not manually specify an ELAN name to join when you configure the host's ELAN driver, the host initially attempts to join the ELAN specified by the LECS. The host successfully joins the ELAN if the LECS is available, the proper LES address for the ELAN has been specified in the LECS, and the LES and BUS are available.

4.2.2.8 Intelligent BUS

This feature reduces broadcast traffic by using the MAC address information in the LES. When an intelligent BUS receives a unicast frame, the BUS first checks the LES's mapping table to see if the MAC address is registered there. If it is, the BUS forwards the frame directly to the destination, instead of broadcasting.

4.3 An Introduction to Multi-Protocol Over ATM

MPOA builds upon the foundation of LANE.

4.3.1 LANE Without MPOA

ATM networks co-exist with and support network applications which may not be ATM-aware. Consequently, ATM protocols are needed to monitor legacy network protocol (IP, IPX, Appletalk, etc.) packets and perform translation into ATM cells and circuits. This monitoring and translation can be performed in one of the following ways:

- in a host protocol stack *after* packet construction and *before* packet transmission
- in a LAN-to-ATM edge device *as* packets move through the network

LANE is one example of such a protocol. It resolves datalink layer addresses into ATM addresses and establishes circuits to the destination addresses. Network addresses *within* a subnet can be learned using LANE's broadcast support.

However, LANE relies on physical routers to deliver packets *across* subnets (see Figure 4.2). Because routers *must* examine – and modify – *every* packet, ATM cells *must* be reassembled into packets, modified, and re-segmented at *every* router hop. This process imposes significant transmission delays between the source and destination of the network traffic.

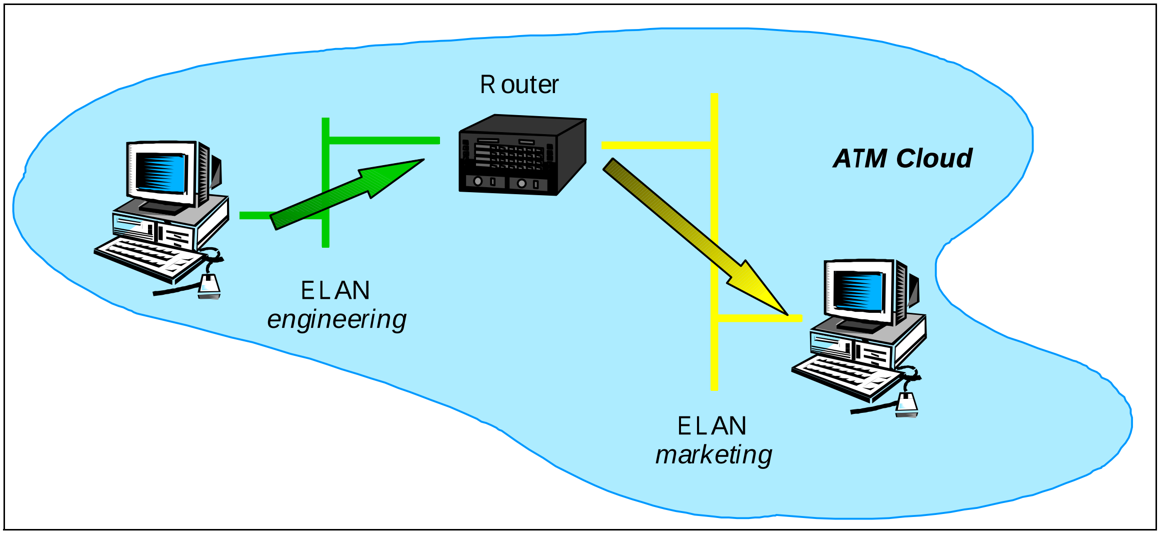

Figure 4.2 - LANE without MPOA

In addition to LANE, protocols such as IP can operate over an ATM network via the IETF Internetworking Over NBMA Networks (ION) Working Group's *Next Hop Resolution Protocol* (NHRP). NHRP allows the ATM network to be divided into Logical IP Subnets (LISs). Using NHRP, routers are still required to interconnect these subnets; but NHRP permits intermediate routers to be bypassed on the *data* path. NHRP allows entities called Next Hop Clients (NHCs) to send queries between different subnets. These queries are propagated using Next Hop Servers (NHSs) via paths found using standard routing protocols. Consequently, NHRP enables the establishment of VCC data paths across subnet boundaries *without requiring physical routers in the data path*.

4.3.2 Why MPOA?

The ATM Forum developed the Multi-Protocol over ATM (MPOA) specification to address the limitations of LAN Emulation. MPOA extends ATM support of legacy networks into the network layer. The main objective of MPOA is the efficient transfer of unicast data between subnet(s).

MPOA introduces LANE/MPOA Clients (LEC/MPCs) and MPOA Servers (MPSs) and defines the protocols that are required for LEC/MPCs and MPSs to communicate. LEC/MPCs issue queries for ATM addresses, and receive replies from the MPS using these protocols. MPOA also maintains interoperability with the existing infrastructure of routers. MPOA Servers reside in routers that run standard Internetwork Layer routing protocols such as OSPF, thus providing integration with existing networks.

ForeThought software implements MPOA shortcuts for IP traffic. It does this by *adding* capabilities to LANE, *not* by replacing LANE. LANE/MPOA client drivers are extended LANE drivers. When handling traffic within the *same* ELAN and subnet, they function like LECs. However, when handling traffic that *crosses* subnets, LEC/MPCs initially work with MPOA servers (MPSs) to use MPS-established hop-by-hop circuits. Then, for traffic flows that exceed configurable limits, *shortcut* circuits are built that allow the traffic to traverse the route *without* the necessity of the router(s):

- reassemble packets from ATM cells
- modify the packets
- and then re-segment the packets for transmission to the next hop

Consequently, traffic flowing through a shortcut VCC moves at essentially wire speed from source to destination (see Figure 4.3).

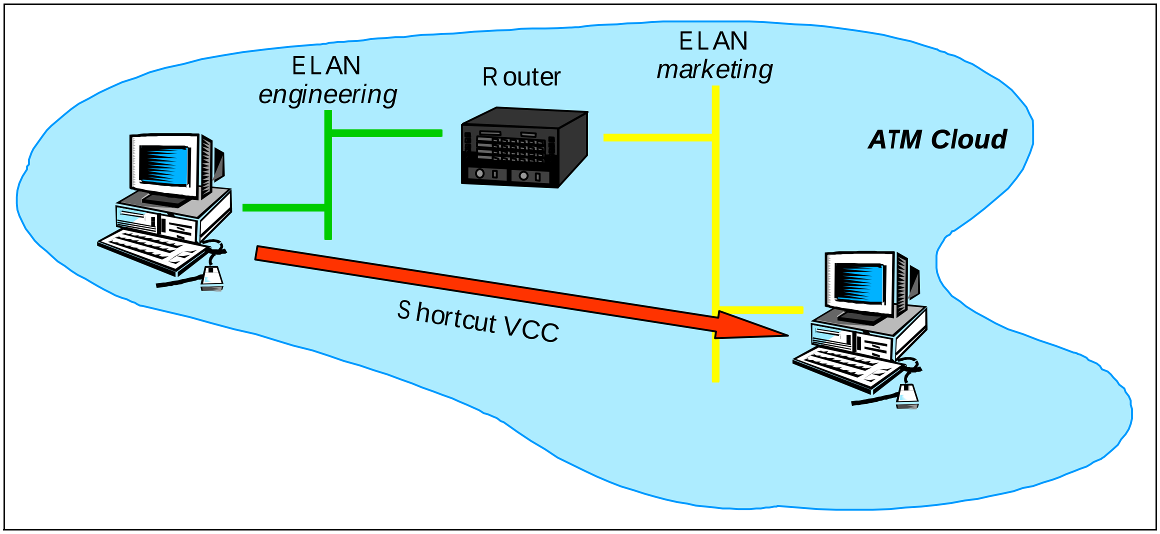

Figure 4.3 - LANE with MPOA

4.3.3 MPOA Components

MPOA requires LANE services for both ELAN traffic handling and MPOA configuration.

LANE/MPOA Client (LEC/MPC)

The LEC/MPC can wear two different "hats." When wearing its LEC "hat," it simply communicates with other ELAN components (the LES and BUS) to resolve MAC addresses into ATM addresses. When it puts on its MPC "hat," the additional function of the LEC/MPC in an MPOA-aware network is to source and sink internetwork shortcut circuits. A LEC/MPC that is the source of a shortcut is known as an *ingress* LEC/MPC. A LEC/MPC that is the sink of a shortcut is known as an *egress* LEC/MPC. The LEC/MPC includes an NHRP Client (NHC).

An ingress LEC/MPC monitors traffic flow that is being forwarded over an ELAN to a router that contains an MPS. When the ingress LEC/MPC recognizes a flow rate (configurable) that could benefit from a shortcut (and thus bypass the routed path), it requests a shortcut to the destination. If a shortcut is available, the ingress LEC/MPC sets up a shortcut VCC, and forwards traffic for the destination over the shortcut.

An egress LEC/MPC receives internetwork traffic from other LEC/MPCs to be forwarded to its local interfaces/users. For traffic received over a shortcut, the egress LEC/MPC adds the appropriate encapsulation and forwards them via a LAN interface (that may be a bridge port, an internal host stack, etc.).

MPOA Server (MPS)

An MPS includes an NHRP Server (NHS) and is the logical component of a router that provides internetwork layer forwarding information to LEC/MPCs. The MPS answers MPOA queries from ingress LEC/MPCs and provides encapsulation information to egress LEC/MPCs.

The MPS also converts between MPOA requests and replies, and NHRP requests and replies on behalf of LEC/MPCs.

4.3.4 MPOA Example

The following are the basic requirements for establishing a shortcut across an MPOA-enabled network:

- There must be LEC / MPCs at each end of the network between which a shortcut is desired.
- The local router interface at each end must be running an MPS.
- A Next Hop Resolution Protocol (NHRP) path must exist between MPSs.

The following example illustrates a typical ATM network that allows MPOA shortcuts to be employed.

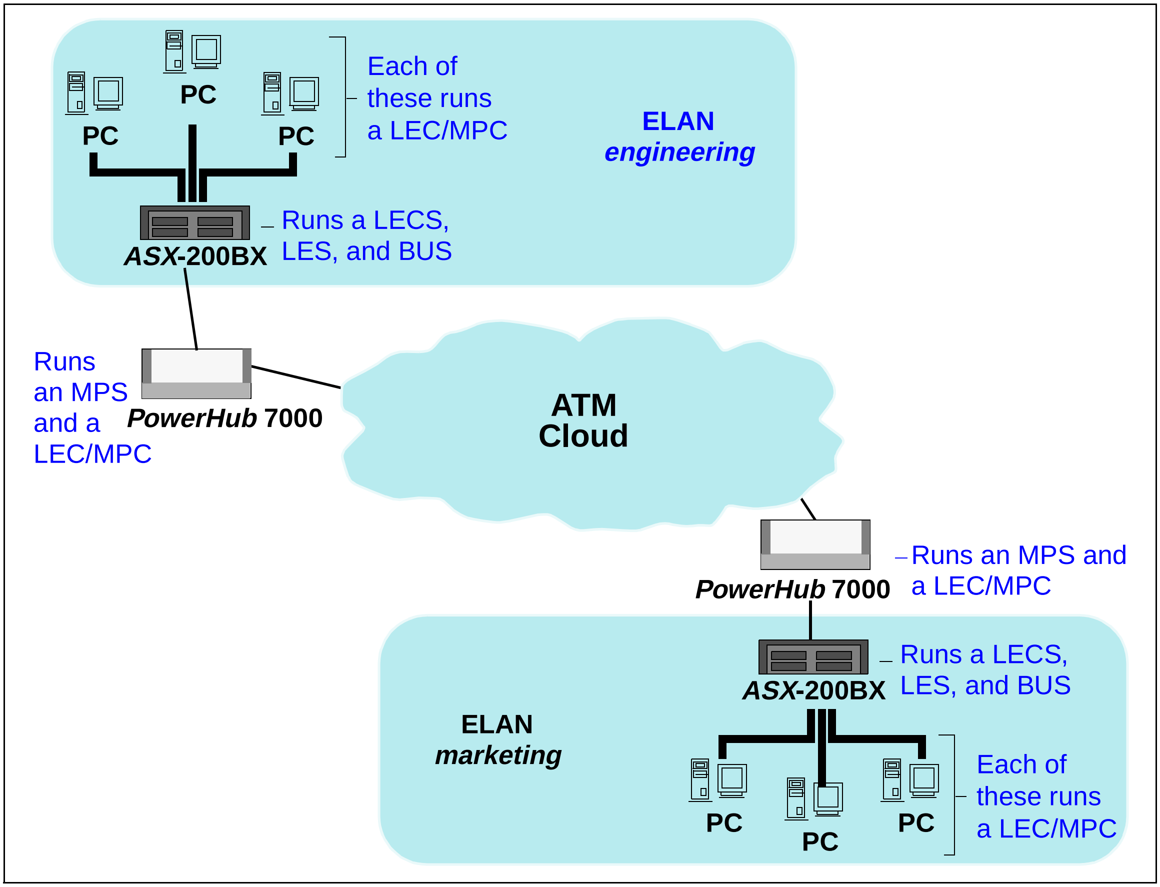
PC
PC
PC
Each of these runs a LEC/MPC
ELAN engineering
Runs a LECS, LES, and BUS
ASX-200BX
Runs an MPS and a LEC/MPC
PowerHub 7000
ATM Cloud
PowerHub 7000
Runs an MPS and a LEC/MPC
Runs a LECS, LES, and BUS
ASX-200BX
ELAN marketing
PC
PC
PC
Each of these runs a LEC/MPC

Figure 4.4 - MPOA Example Network

4.3.4.1 MPS Configuration

The network administrator must configure each MPS with the site-specific IP address matching the gateway address being used by LEC / MPCs in its ELAN.

The MPS on each *PowerHub* 7000 is configured as follows:

1. For each LANE / MPOA virtual port, specify an ELAN name. The LECS configuration must also be updated to allow the MPS to join these ELANs.
2. For each LANE / MPOA virtual port, specify an IP address.
3. Enable LANE / MPOA support.
4. Enable routing.
5. Save the configuration and reboot the MPS if necessary to make the changes effective.

Router table information need not be configured. The MPS will instead gather this information through routing protocol exchanges with other MPS's.

4.3.4.2 Initialization

When its host boots, *each* LEC / MPC automatically goes through the following sequence to establish a connection to the MPS.

1. The LEC / MPC registers via ILMI with the switch to which it is attached.
2. The LEC / MPC connects to an LECS to which it sends its own ATM address and the name of the ELAN it wishes to join (the ELAN name is an empty string unless the LEC / MPC has been site-configured with an ELAN name). The LEC / MPC also supplies a LANE 1.0 compliant parameter identifying itself as an MPOA-aware client.
3. Next, the LEC / MPC receives the following from the LECS:
   - the name of the ELAN to which it is assigned
   - the ATM address of the LES for the ELAN it is joining
   - the parameters containing the flow detection and shortcut establishment policies it is to use
4. The LEC / MPC then connects to its assigned LES, and provides the LES with a parameter identifying itself as MPOA-aware.
5. Finally, the LEC / MPC connects to the ELAN's BUS.

Once the LANE / MPOA connections are established, third-party network-layer protocol drivers on the host can establish network-layer connectivity. The methods these upper-layer drivers use to determine host IP addresses, default gateway, and backup gateway addresses vary depending on the third-party product. For example, the LANE / MPOA driver itself permits these drivers to use BOOTP or DHCP to obtain IP configuration information.

4.3.4.3 Flow Analysis

On a LEC / MPC's host or edge device, IP packets with destinations *within* the host's subnet are sent using LANE 1.0 methods; i.e., the client puts on its LEC "hat" and works directly with its ELAN's services to connect with local destinations. Packets destined for *remote* subnets cause the LEC / MPC to put on its MPC "hat." This client is then referred to as an "ingress LEC / MPC."

Ingress LEC / MPCs associate destination IP addresses with shortcut circuits. Ingress LEC / MPCs use configurable parameters called *flow descriptors* to determine *whether* and *when* to trigger creation of shortcut circuits. The ingress LEC / MPC also monitors the most recent use of a shortcut circuit to determine when to tear down the shortcut. Specifically, when an ingress LEC / MPC sends a packet:

1. If a shortcut circuit *already exists* to the IP destination, the LEC / MPC sends the packet over this circuit.
2. If no shortcut circuit exists, the LEC / MPC determines *whether* shortcuts to this IP address are allowed. If shortcuts to the destination IP address are *not* allowed, the LEC / MPC sends the packet to the gateway router.
3. If no shortcut circuits exist, *and* shortcuts to the IP address *are* allowed, the LEC / MPC determines if the packet traffic flow exceeds the shortcut enable trigger value (set by the flow descriptors) for the destination IP address's flow. If the flow exceeds the trigger value, the LEC / MPC tries to establish a shortcut circuit to the destination LEC / MPC (called the egress LEC / MPC). If the flow does not exceed the trigger value, the ingress LEC / MPC simply sends the packet traffic to the gateway router.

4.3.4.4 Making a Shortcut

When the ingress LEC / MPC determines that the packet traffic flow exceeds the shortcut-enable trigger value, the ingress LEC / MPC tries to establish a shortcut circuit to the egress LEC / MPC. The following describes how a shortcut is set up:

1. The ingress LEC / MPC initiates the shortcut creation process by sending an MPOA resolution request to the MPS it uses as a gateway router (this MPS is called the *ingress* MPS). The MPS converts the request to a next hop resolution protocol (NHRP) request. This NHRP request includes the destination's IP address and asks for the corresponding ATM destination address.
2. This request is passed along hop-by-hop until it reaches the final MPS (called the *egress* MPS) on the route to the destination IP address.
3. The egress MPS sends a cache imposition request to the egress LEC / MPC. The egress LEC / MPC sends a cache imposition reply, which is converted to a NHRP response by the egress MPS. The ingress MPS converts the NHRP response to an MPOA resolution response and transmits it to the ingress LEC / MPC.

4. When the ingress LEC / MPC receives the NHRP response containing the destination's ATM address, it first checks if a shortcut circuit to that ATM address already exists. If a shortcut circuit to that address already exists, it sends the packets via the existing shortcut circuit. If no shortcut circuit exists it opens a new shortcut circuit and begins sending packets over it to the destination.

4.3.4.5 Shortcut Teardown

Application programs and networking protocol stacks are MPOA-ignorant and, therefore, do not tear down shortcut circuits when the shortcut is no longer needed. Therefore, the MPOA layer itself tears down seldom-used shortcuts to avoid circuit exhaustion in the client and network. When a shortcut is idle for a period exceeding a set limit, the shortcut is torn down.

CHAPTER 5 *ForeThought* PNNI

PNNI (Private Network Node Interface or Private Network-to-Network Interface) is an ATM Forum approved protocol which defines interoperability between private ATM switches. PNNI defines both the routing and signalling standards for inter-switch interoperability.

This chapter provides an overview of FORE Systems' pre-standard version of PNNI, *ForeThought* PNNI (FT-PNNI), and its use in a multiple-switch network. FT-PNNI is a scalable routing and signalling protocol used in networks containing multiple FORE switches. FT-PNNI simplifies large network topologies by organizing the nodes (switches) in that network into smaller groups.

It is this reorganization of the network topology that makes FT-PNNI's simplified routing possible. By segmenting a large network into smaller peer groups of nodes, FT-PNNI reduces the amount of network topology information that those very nodes must maintain.

5.1 FT-PNNI Routing

The FT-PNNI routing protocol serves to distribute topology and address reachability information between switches and groups of switches in a network. This topology and addressing information is used by switches to compute paths through the network. The functions of the FT-PNNI routing protocol include the following:

- Hello Protocol
- Topology Database Exchange
- Flooding
- Hierarchical Routing

5.1.1 Hello Protocol

FT-PNNI hello packets are exchanged between neighboring nodes. Each switch (node) transmits a hello indication on each of its FT-PNNI routing channels at regular intervals. The time between these hello indications is called the Hello Indication Interval. When a switch receives a hello indication from one of its neighbors, it stores the logical link (loglink) from that neighbor to itself in the topology database.

Based on the hello indications, loglinks are refreshed periodically. Since loglinks discovered as a result of hello indications are unidirectional, each switch stores unidirectional loglinks with its immediate neighbor as the source and itself as the destination.

5.1.2 Topology Database Exchange

Each switch sends to each of its neighbors a group of loglinks from its topology database at regular intervals, called the NSAP map indication interval. This exchange of information between neighboring switches ensures that the topology databases of the switches stays synchronized.

5.1.3 Flooding

Flooding is a reliable hop-by-hop propagation of loglinks throughout a peer group. Whenever a new loglink is discovered by a switch, the switch immediately broadcasts the existence of this link to all of its neighbors. The neighboring switches then broadcast the existence of the same link to all of their neighbors. Very quickly, the existence of the new loglink is flooded throughout all of the switches in the network.

Similarly, when a link goes down, or when a *significant* change is seen in the metrics of a loglink, the latest state of the loglink is propagated immediately throughout the network.

5.1.4 Hierarchical Routing

FT-PNNI operates in a hierarchical topology. The structure of the hierarchy is defined by the peer group ID in a routing domain. Address assignment in FT-PNNI, therefore, corresponds to this hierarchical topology, providing increased scalability.

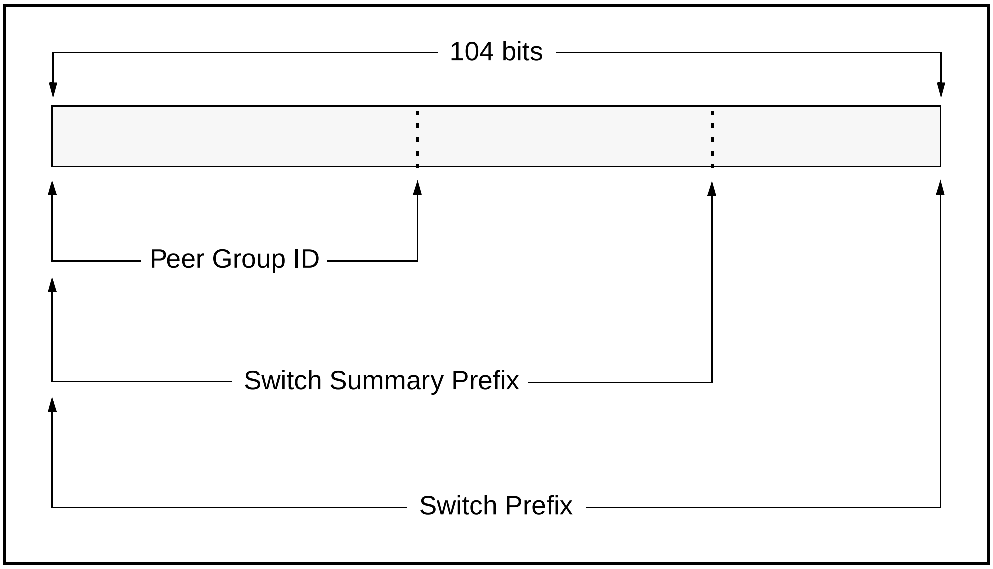

Figure 5.1 - Example of a 13-byte Switch Prefix

5.1.4.1 Hierarchical Addressing

FT-PNNI uses private ATM address prefixes (NSAP prefixes) as node identifiers. FT-PNNI does not distinguish between node identifiers and reachability information. Thus, the IDs of nodes in the FT-PNNI addressing map are NSAP prefixes. In the default case at the lowest level peer group, the switches have a 13-byte prefix as their node ID and end systems (hosts) have a 19-byte prefix as their node ID.

5.1.4.1.1 Switch Prefix

Each switch in a FT-PNNI network is configured with a 13-byte prefix called the switch prefix. Hosts that are attached to the switch are presented with this prefix during ILMI address registration. In this way, end systems are configured with a private ATM address that includes the 13-byte switch prefix.

5.1.4.1.2 Switch Summary Prefix

Each switch is configured with a switch mask (*swmask*) which gives the length of the switch summary prefix within the switch prefix. The *swmask* gives the number of most significant bits of the switch prefix that constitute the switch summary prefix. Since all end system addresses attached to a switch have the same switch summary prefix, their reachability information can be summarized by this prefix (i.e., by the switch summary prefix).

5.1.4.1.3 Peer Group ID

Each switch is configured with a peer group mask (*pgmask*) which gives the length of the peer group ID within the switch summary prefix. The *pgmask* gives the number of most significant bits of the switch summary prefix that constitute the peer group ID.

Each peer group has a peer group ID that uniquely identifies it from every other peer group. Every node (switch or end system) in a particular peer group shares that same unique peer group ID, thereby indicating membership to that peer group.

A simple example of summarizing by peer group ID can be seen in Figure 5.3, where every switch (and end system, although not shown) in peer group A is identified starting with “A.”

5.1.4.2 Path Computation

Path computation is performed on demand whenever FT-PNNI signalling requests a path to a given destination. The Bellman-Ford Shortest Path algorithm is used to compute the shortest path tree of all nodes in the topology with the local node as the source. The administrative weight metric in the loglinks is used as the minimizing criterion in computing the shortest path route. In the case of finding multiple equal-cost paths to a given destination, available cell rate is used to break ties.

5.2 The Physical Network

In an ATM network, data is sent and received over virtual circuits, or circuits that only exist when needed. This communication over these virtual circuits is made possible by signalling that occurs between the switches in the network.

In a network of FORE switches, any new addition to the topology is recognized immediately by all nodes (switches) having a direct connection to the new node. Then each switch that has recognized the new switch sends a message to each of its direct connections, and so on. Eventually, and within a very brief period of time, every switch in the network is aware of the new addition and the links by which that new addition can be reached. This topology is stored by each switch in its local topology database.

In a small, local area network (LAN), the topology is relatively simple, meaning that the switches in the LAN have a relatively small topology database to maintain. As the LAN grows, however, and more switches are added, that same database can grow to be very large in a short period of time.

As this topology database grows, the amount of information a switch must look up when searching for an address also grows. In the end, this can result in delayed connection set-up, congestion in the network, and even lost data.

Figure 5.2 depicts a typical ATM network, containing 21 FORE switches (⊠). The hierarchy of this network is flat, meaning that each switch must be aware of all the other switches in the network, as well as all the possible routes to those switches. As more switches are added to this network, the hierarchy will become more complex and the switches will have to contend with larger topology databases.

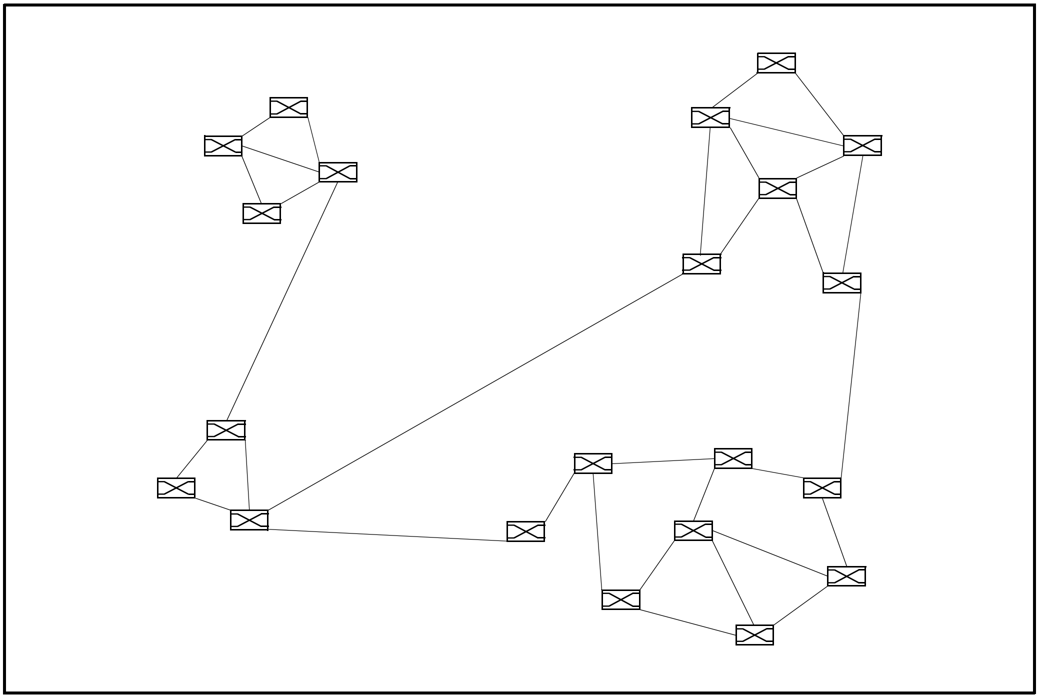

Figure 5.2 - Private ATM Network with 21 Switches and 34 Bidirectional Links

It is in these large, single-level networks that FT-PNNI is most useful, because it lets you simplify large network topologies by creating a two-level hierarchy. In this hierarchy, clusters of contiguous switches are grouped together and they are collectively summarized by a single, logical node.

Figure 5.3 shows the same network as in Figure 5.2 after being organized into peer groups, now having a two-level hierarchy. The subsections that follow explain the organization of these peer groups, how they simplify the overall network topology, and how they change the logical view of the network.

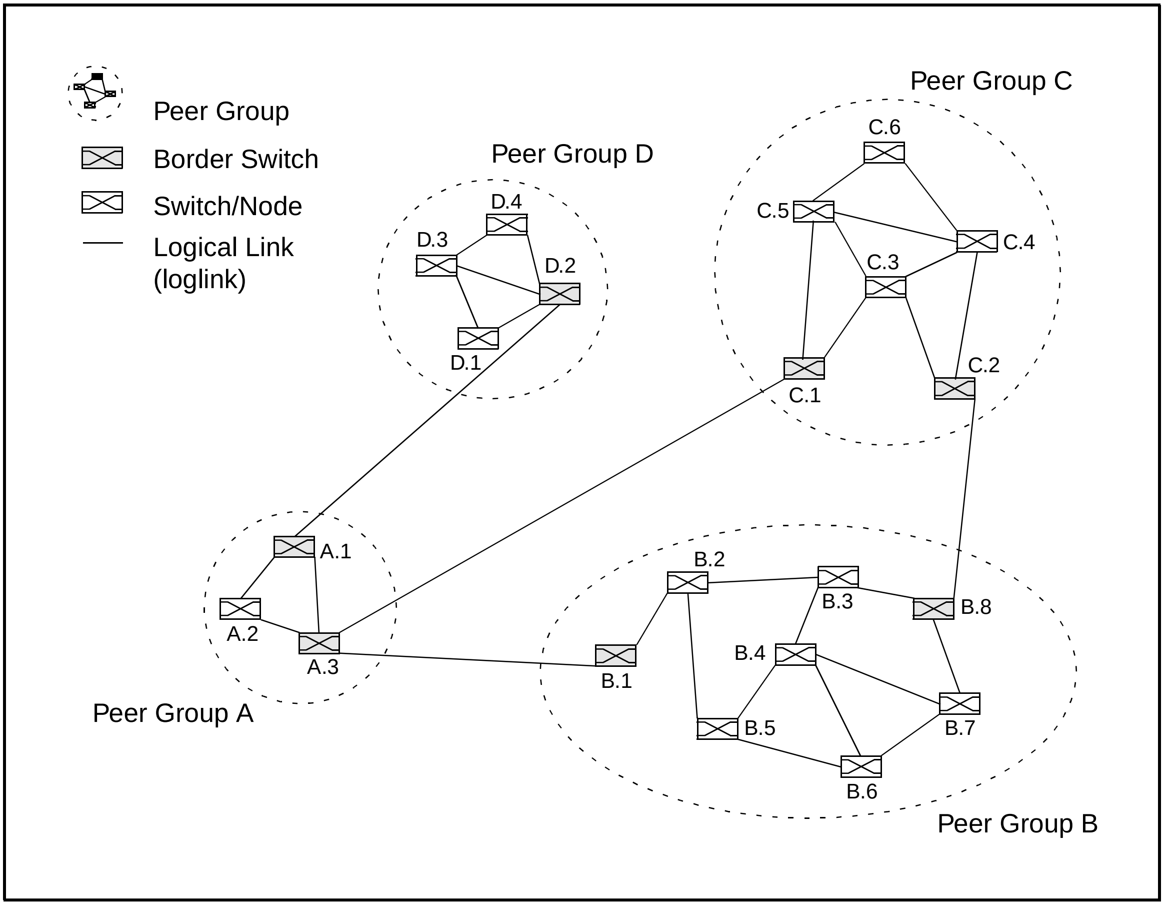

Figure 5.3 - Example of FT-PNNI Hierarchy Showing Lowest-Level Peer Groups

5.2.1 Peer Groups

The FT-PNNI hierarchy begins with a network of switches, organized into peer groups. A peer group is a collection of interconnected switches that are organized into a group. Peer group organization can be determined by a network administrator, but switches that are located close to one another are usually made into a peer group.

The network shown in Figure 5.3 is organized into four lowest-level peer groups: A, B, C, and D. The switches within a certain peer group are numbered according to that particular group. For example, the switches in peer group A are identified as A.1, A.2, and A.3.

Peer groups have a peer group identifier (ID), assigned at configuration time and exchanged in hello indication messages. Neighboring switches can determine if they belong to the same peer group by comparing these peer group IDs.

Switches in a peer group are aware of the topology of their own peer group and the existence of all other peer groups. They recognize the links between their peer group and others, but they are not aware of internal topology information of other peer groups. Instead, the topologies of other peer groups are summarized as a single, reachable location, known as a peer group summary node (PGSN).

5.2.2 Peer Group Topology

To maintain an accurate and updated view of its relative location and status, a switch periodically sends a hello indication message to every other switch with which it has a direct connection. These hello indications contain the switch prefix, peer group membership information, and link metrics (attributes) for the physical link between the two switches.

Through this regular exchange of messages, each switch learns which switches are its immediate neighbors, to what peer groups they belong, and whether or not the link between itself and its neighbors is valid.

5.2.3 Border Switches

A border switch is any switch that has at least one link to a switch in another peer group. Border switches play an important role in FT-PNNI because they are responsible for summarizing reachability information for their respective peer groups, appropriately filtering the flow of topology database information across peer group boundaries, and building the lowest level source route for call setups entering the peer group.

5.2.4 Peer Group Summary Node (PGSN)

A PGSN is a virtual (logical) or imaginary node that summarizes a peer group’s reachability information. The PGSN has the peer group ID of its peer group as its switch summary prefix. Each border switch in the peer group advertises a logical link (loglink) to the PGSN. The PGSN is a logical representation of the switches contained in a peer group.

5.2.5 Backbone Topology

Loglinks between border switches and loglinks from border switches to PGSNs are called backbone links and considered part of the backbone topology. Information regarding these backbone links is propagated across peer group boundaries during database exchange and flooding.

5.2.6 Single Switch Perspective

The main reason for grouping switches in large networks is to simplify each individual switch’s view of the topology. For example, each switch in peer group A is aware of every other switch in peer group A, the border switches in the rest of the network, the links between them, and the backbone topology. Switches in peer group A are not aware, however, of the internal topology of other peer groups. Instead, individual switches see a PGSN (see Figure 5.4).

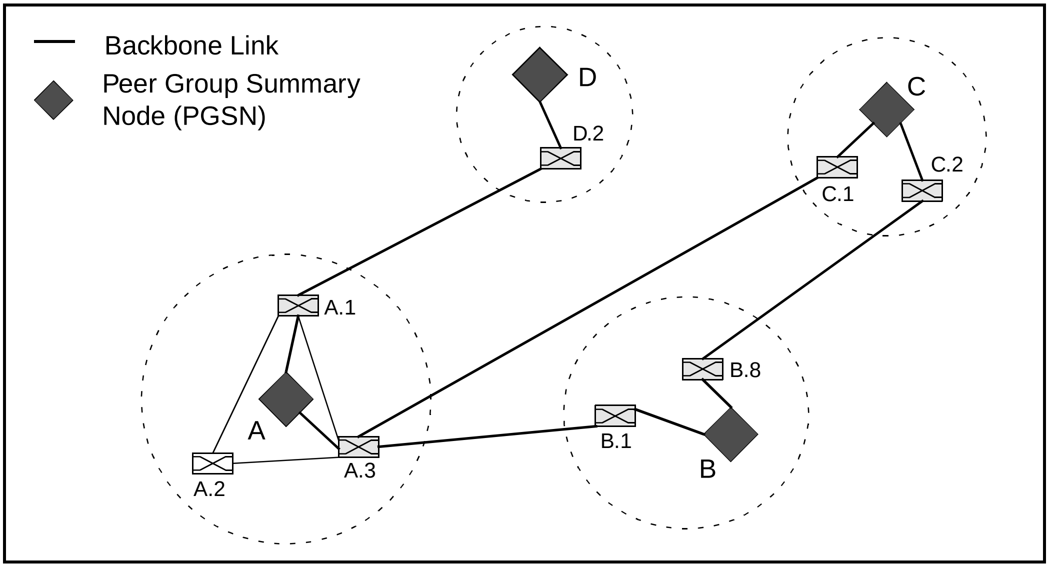

Figure 5.4 - View of the Network from Switches in Peer Group A

CHAPTER 6 ATM Forum PNNI

In April, 1996, the ATM Forum (ATMF) approved an industry specification for PNNI Phase 1 (Private Network Node Interface or Private Network-to-Network Interface). PNNI (Private Network Node Interface or Private Network-to-Network Interface) is a scalable protocol that defines both the routing and signalling standards for interoperability between private ATM switches. PNNI simplifies large network topologies by organizing the nodes in a network into smaller groups. Through this re-organization, PNNI reduces the amount of topology information that those nodes must maintain.

The ATMF PNNI 1.0 Specification requires every PNNI implementation to support peer group construction and communication within a peer group; support of communication among peer groups using border nodes, split switches, or PGL hierarchy is optional.

ForeThought 5.x supports the mandatory requirements defined in the ATMF PNNI 1.0 Specification, includes some of the defined optional requirements, and includes FORE extensions to provide improved calls-per-second performance, easier configuration, configurable address filtering, and scalability to larger networks. *ForeThought* 5.x also allows you to connect peer groups to each other using the split-switch feature, but it does not support the optional ATMF border node or Peer Group Leader (PGL) hierarchy features.

ForeThought 6.x continues the support of the features mentioned above and adds support for the optional ATMF border node subset, the PGL/Logical Group Node (LGN) subset, and the border node with LGN subset.

This chapter provides an overview of FORE Systems' implementation of ATMF PNNI (hereafter referred to as PNNI), and its use in a multiple-switch network. The following topics are discussed in this chapter:

- **Section 6.1 -** PNNI Routing Protocol
- **Section 6.2 -** PNNI Signalling Protocol
- **Section 6.3 -** Border Node, PGL, and LGN Support
- **Section 6.4 -** Internetworking between PNNI and FT-PNNI
- **Section 6.5 -** Path Extension QoS
- **Section 6.6 -** Load-Balanced Routing
- **Section 6.7 -** Configuring PNNI

6.1 PNNI Routing Protocol

The PNNI routing protocol enables the distribution of topology and address reachability information between switches and groups of switches. *Reachability information* is the data that explains how to contact a single ATM address or a group of ATM addresses that are summarized by a single prefix. This information is used to compute routes across the network. The key elements of the PNNI routing protocol are as follows:

- **Section 6.1.1 -** Hello Protocol
- **Section 6.1.2 -** Database Exchange Protocol
- **Section 6.1.3 -** Flooding Protocol
- **Section 6.1.4 -** Path Computation
- **Section 6.1.5 -** Hierarchical Routing

6.1.1 Hello Protocol

The Hello protocol is used to discover and verify the identity of neighbor nodes and to determine the status of the logical links to those nodes. Logical links between nodes in the same peer group are called *horizontal links*. Hello packets are exchanged at regular intervals over links connecting neighboring nodes. Initially, these packets are used to exchange identities between the nodes and to determine if they belong to the same peer group. After that, Hello packets are used as keep-alive messages for the links over which the packets are exchanged. After a configured number of hello intervals have passed without receiving a hello message, the link is declared to be down.

6.1.2 Database Exchange Protocol

When neighboring nodes determine that they are peers; i.e., that they belong to the same peer group and they are running compatible versions of the PNNI protocol, they begin a database exchange process. At regular intervals, each node tells the other what information it has in its topology database and requests similar information from the other neighboring nodes. This exchange of information ensures that the topology databases of all switches in the network stay synchronized.

6.1.3 Flooding Protocol

Topology information is organized into smallest units called PNNI Topology State Elements (*PTSEs*). PNNI routing uses a flooding mechanism to propagate these PTSEs throughout the network in a hop-by-hop fashion. PTSEs are encapsulated into PNNI Topology State Packets (*PTSPs*) for transmission. PTSPs always contain the node identifier (ID) and the peer group ID of the switch that originated all of the PTSEs.

When a PTSP is received, its component PTSEs are examined. Each PTSE is acknowledged by sending a PTSE acknowledgment packet (containing possibly a bundle of PTSE acknowledgments) to the sending peer. If the PTSE is more recent than the copy that the receiving node has in its topology database, the database copy is updated and is re-flooded to all neighboring peers except the one from which the PTSE was received. A PTSE sent to a neighboring peer is periodically retransmitted until it is acknowledged.

In this fashion, when a link is created or goes down, this information is propagated immediately throughout the network. Similarly, when a *significant* change is seen in the metrics of a link, this state change is flooded throughout the network.

6.1.4 Path Computation

Path computation is done on the source (or originating) switch. All of the network topology information obtained and updated by the higher layers of the protocol is used to determine a path through the network which is efficient and which satisfies all of the relevant Quality of Service (QoS) parameters.

The path is fully specified by the source switch all the way to the destination. Path information is stored in a DTL, which is forwarded along with the setup message to establish the circuit for a particular call.

6.1.5 Hierarchical Routing

Hierarchical routing allows contiguous switches to be organized into peer groups. The switches within each peer group exchange detailed topology information about their own group that is not visible to switches outside the peer group. Similarly, switches within each peer group do not receive detailed topology information about switches outside of their peer group. Instead, each peer group is represented externally by its Peer Group Leader (PGL). The PGLs exchange among themselves summary topology and reachability information pertaining to their respective peer groups.

This idea is illustrated in Figure 6.1. As you can see, the switches in a given peer group know about the topology information of their own peer group. However, they are not aware of the individual members of the other peer group. They only know how to reach the other group via their PGL.

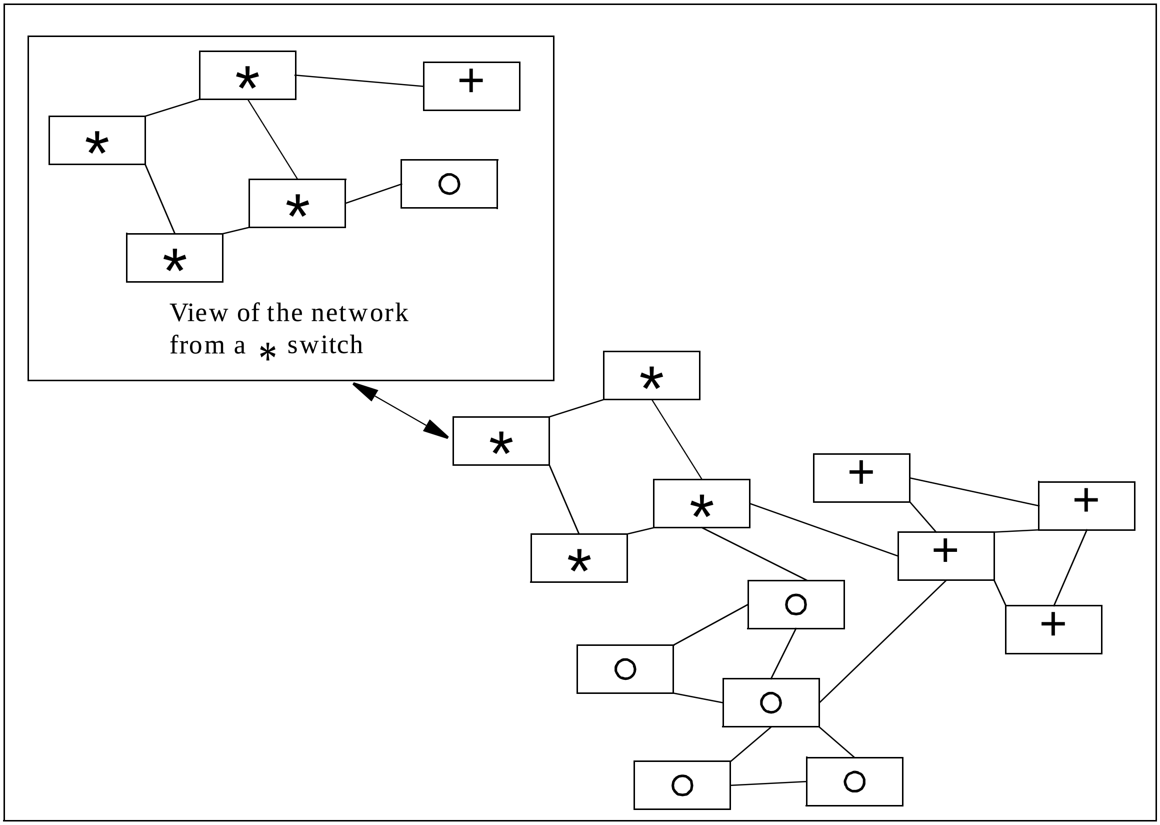

Figure 6.1 - Peer Groups Summarized into a Single Address

Hierarchical routing reduces the number of links that each switch must maintain. For example, if the switches in Figure 6.1 were in a flat network, each switch would have to be aware of all 16 links shown. If the switches are separated into peer groups, the routing table for each switch becomes smaller.

In the example shown in the inset in Figure 6.1, a * switch only needs to know about the four links inside its peer group plus one link to the O peer group and one to the + peer group. This is a total of six links, as opposed to 16. The example in the inset is hierarchical because the * switches view all of the O switches as a single node and all of the + switches as another single node.

ForeThought 6.x supports the dynamic configuration of ATMF PNNI PGL hierarchical networks. (Although *ForeThought* 5.x does not support the dynamic configuration of ATMF PNNI PGL hierarchical networks, FORE switches running *ForeThought* 5.x can be placed in and interact appropriately within PNNI hierarchical PGL networks.) For more information about ATMF border node capabilities, ATMF PGL/LGN capabilities, and ATMF border node with LGN capabilities, see Section 6.3. For more information about how *ForeThought* can be configured to connect multiple ATMF PNNI peer groups, see Section 6.4.1.

6.1.5.1 Peer Groups

A *peer group* is a group of physically connected nodes that have a portion of their NSAP address in common. As shown in the example in Table 6.1, the switches share the first portion of their NSAP prefix.

Table 6.1 - Example of Peer Group Addressing

Peer Group A	NSAP Prefix
switch_1	47.0005.80.ffe100.1111.1111.21b8
switch_2	47.0005.80.ffe100.1111.1111.82c3
switch_3	47.0005.80.ffe100.1111.1111.1559
switch_4	47.0005.80.ffe100.1111.1111.aa17

All of the NSAP addresses of the group members can be represented as a single, shortened (summarized) address. Additionally, every member of a peer group has the same level. Members of a peer group are connected to each other by horizontal links.

Each peer group is summarized by its PGL as a single group node. A *node* is a logical entity that resides in a switch and performs routing operations such as discovering other nodes in the network, maintaining a topology database of its peer group, exchanging that database with its neighbors, and computing paths from itself to other nodes in the network.

6.1.5.2 PGL/LGN Hierarchy

As shown in Figure 6.2, each peer group elects one PGL. (See Section 6.3.8.2 for more information about PGL elections.) Upon being elected, the PGL activates a LGN within the same switch. The LGNs represent the peer group at the next higher (parent) level.

The interconnection of LGNs form a *higher-level topology.* LGNs can, in turn, be grouped into higher-level peer groups that, in turn, are represented by LGNs in an even higher level topology, and so on. LGNs communicate with each other over dynamically established SVCC RCCs. The LGNs summarize information about the child peer groups they represent and flood topology information learned at the higher level to the PGL. The PGL, in turn, floods that information to its peer group.

For example, as shown in Figure 6.2, switch B.1.1 belongs to the lowest-level peer group B.1, which, in turn, belongs to a higher-level peer group B. Switch B.1.1 has detailed topology information of peer group B.1, as well as higher-level topology information of peer group B. Peer groups B.2 and B.3 also belong to peer group B. However, switch B.1.1 knows about the LGNs representing B.2 and B.3, respectively, although it does not know the detailed topology inside peer group B.2 or peer group B.3.

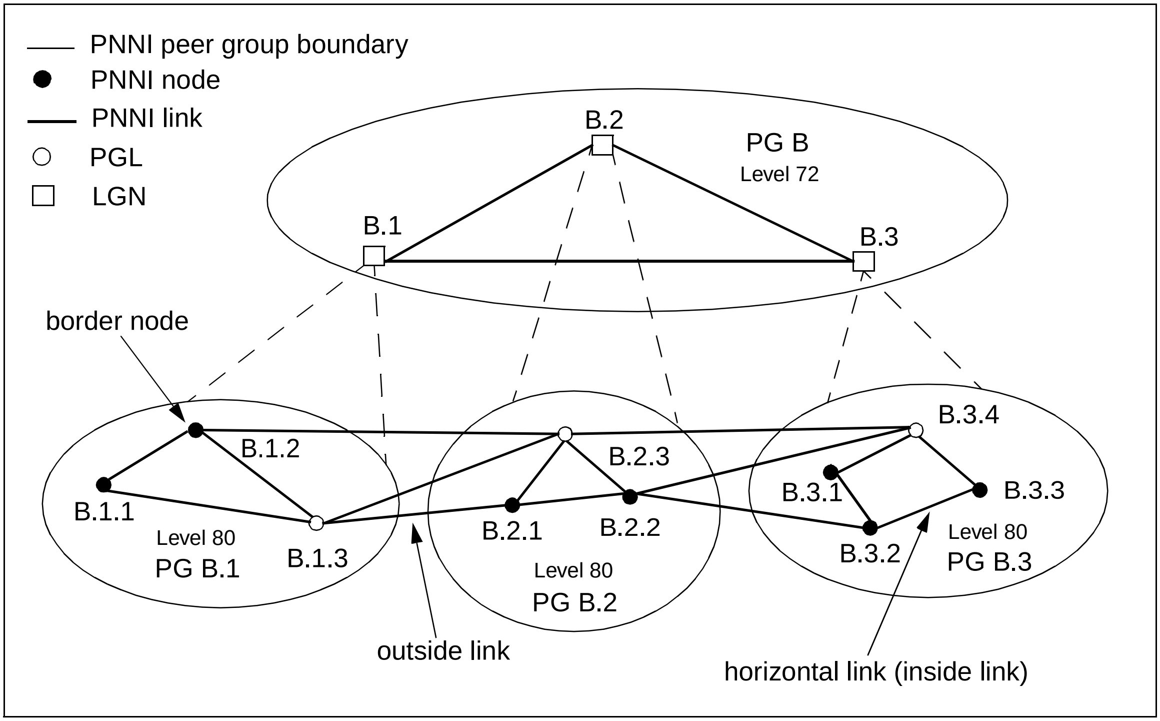

Figure 6.2 - PGL Hierarchy Example

6.1.5.3 Border Nodes

Border nodes, such as switch B.1.2 in Figure 6.2, have at least one link that crosses a peer group boundary (an outside link). No flooding takes place on the outside links. The border nodes only use the outside links for hello messages. The hello messages advertise reachability across the border. In Figure 6.2, switch B.1.2 advertises that it can reach B.2, while switch B.2.3 advertises that it can reach B.1.

6.1.5.4 Lop-sided Hierarchy

A lop-sided hierarchical network is one in which a lowest level node discovers that it is in a peer group with one or more of its neighbors' ancestors; i.e., the lowest level node is acting as a peer to one or more LGNs. As shown in lop-sided hierarchy example in Figure 6.3, switches B.4, B.5, and B.6 are all lowest level nodes. However, they are part of peer group B which contains parent node LGNs B.1, B.2, and B.3.

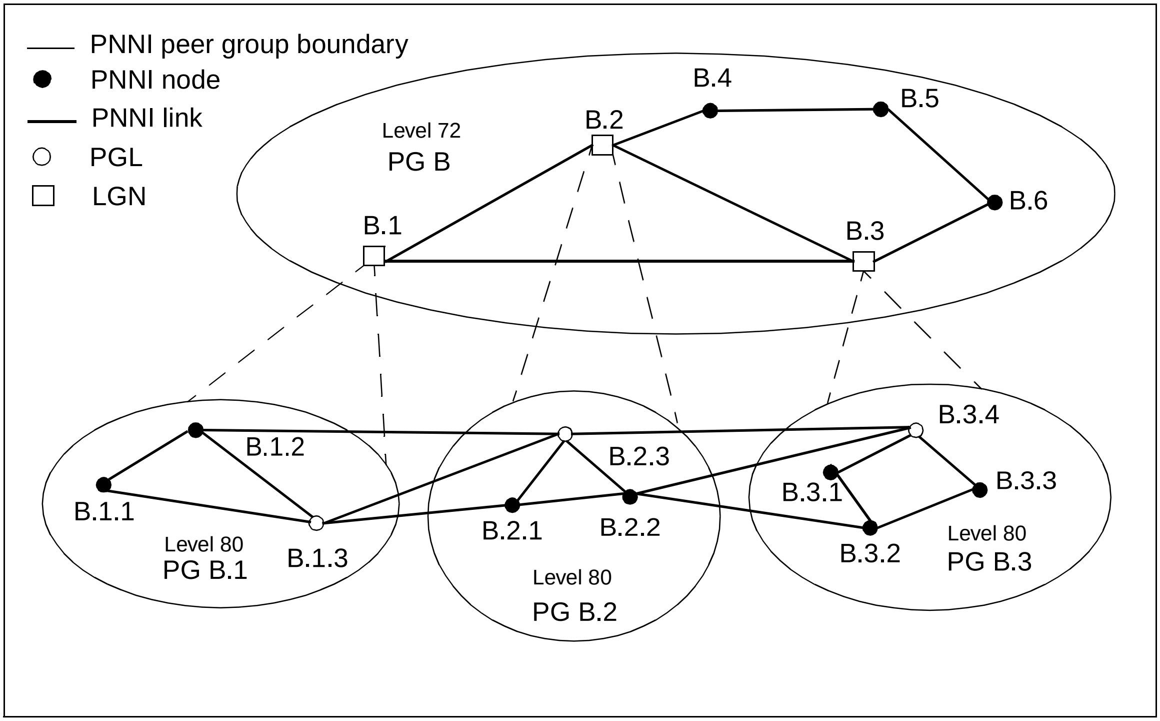

Figure 6.3 - Example of a Lop-sided Hierarchy

6.2 PNNI Signalling Protocol

PNNI signalling is used to establish point-to-point and point-to-multipoint connections across the network. This protocol is based on ATMF UNI signalling with mechanisms added to support source routing and crankback.

6.2.1 Source Routing and DTLs

A directed Designated Transit List (DTL) is a source route which specifies the preferred call routing path that the PNNI router should use when setting up an SVC or an SVP. Each DTL is a source route and each entry in the DTL represents a single hop in that source route. Each hop is represented by a PNNI node and the logical output port at that node.

ATM Forum PNNI DTLs are created using the **conf atmr pnni dtl new** command as described in Part 1 of the *AMI Configuration Commands Reference Manual*. A DTL is a source route (uniquely identified by a pair of indices: node index and DTL index) which specifies the preferred call routing for the SVC portion of a directed SPVC and the SVP portion of an SPVP. Each entry (row) in the DTL represents a single hop in the source route.

Once you have created the DTLs, you must create a table (list of DTLs) using the **conf atmr dtltable new** command. In this table, each DTL is assigned a weight which indicates the priority in which the DTLs are tried, with the highest weight being the first one tried. The weights are arbitrary (e.g., 10, 20, 30, and 40). You can even assign two or more DTLs with the same weight. The table is also assigned an arbitrary tag number which uniquely identifies the table. This tag can be applied on a per-SPVC or per-SPVP basis using the **-dtltag** advanced option under **conf spvx spvcc pnni pp new** (or **modify**) or under **conf spvx spvpc new** (or **modify**). Multiple SPVCs or SPVPs can use the same DTL table.

When an SPVC or SPVP is created with a DTL tag assigned to it, the DTL in the corresponding table with the highest weight is tried first. If this DTL is unusable, the DTL with the next highest weight is tried. This process is repeated until a DTL in that table is found that provides a route to the destination which satisfies the QoS requirements.

If none of the user-specified DTLs in the table can be used, and if the **-autoDtl** option under **conf spvx spvcc pnni pp new** (or **modify**) is set to **enable**, the call is dynamically routed if an alternative path can be computed. If that option is set to **disable**, then the call setup fails, but the DTLs are retried periodically.

If any of the DTLs fail, an entry is made in the syslog which indicates that the attempt failed, which DTL was tried, and the time at which the failure occurred. When a DTL fails, a back-off mechanism is used to determine when to retry that particular DTL. The DTL is not attempted for a user-configured interval in order to allow any old information to age out and allow the network to stabilize. This interval is set under **`conf spvx spvcc pnni parameters backoff interval`** *<seconds>* for SPVCs. It is set under **`conf spvx spvpc parameters backoff interval`** *<seconds>* for SPVPs.

However, you can enable or disable this back-off mechanism on a per-SPVC or per-SPVP basis. Use the **`-backoff`** option under **`conf spvx spvcc pnni pp new`** (or **`modify`**) for SPVCs. Use the **`-backoff`** option under **`conf spvx spvpc new`** (or **`modify`**) for SPVPs. The default setting for both SPVCs and SPVPs is **`enable`**. If you disable this mechanism, then instead of waiting for the user-configured back-off interval after a failure, the DTLs are retried at every reroute interval. If the **`-backoff`** option is then changed to enable using the **`modify`** command on an existing SPVC or SPVP, the remaining waiting period for any failed DTLs is cleared and the DTLs are attempted again.

6.2.2 Crankback

During PNNI signalling, a call being processed according to a DTL may encounter a blocked node or link along the designated route. Crankback allows a partial reroute of such a rejected call so that it does not have to be cleared the whole way back to the source. Additionally, an indication of the blockage and subsequent reroute information is sent to the originator of the DTL.

6.3 Border Node, PGL, and LGN Support

ForeThought 6.x supports the ATMF border node subset, the PGL/LGN subset, and the border node with LGN subset of the ATMF PNNI Specification. A border node is a logical node in a specified peer group which has at least one link which crosses the peer group boundary. The border node subset provides the following capabilities:

- **Section 6.3.1 -** Outside Links and Hello Protocol over Outside Links
- **Section 6.3.2 -** Origination of Uplinks and Uplink PTSEs
- **Section 6.3.3 -** Entry and Exit Border Node DTL Handling
- **Section 6.3.4 -** Entry Border Node Crankback Handling

A PGL is a node in a peer group that collects and assimilates data to represent the entire peer group as a single node in the form of a parent node. An LGN is a virtual representation of a lower level peer group as a single node in terms of one level of the PNNI routing hierarchy. The PGL/LGN subset provides the following capabilities:

- **Section 6.3.2 -** Origination of Uplinks and Uplink PTSEs
- **Section 6.3.5 -** SVCC RCC Establishment
- **Section 6.3.6 -** SVCC RCC Hello Protocol
- **Section 6.3.7 -** LGN Horizontal Link Hello Protocol and PTSEs
- **Section 6.3.8 -** PGLs
- **Section 6.3.9 -** Aggregation Tokens and Link Aggregation
- **Section 6.3.10 -** Nodal Aggregation
- **Section 6.3.11 -** Internal and Exterior Address Summarization
- Anycast

Similarly, the border node with LGN support subset provides the following capabilities:

- **Section 6.3.5 -** SVCC RCC Establishment
- **Section 6.3.6 -** SVCC RCC Hello Protocol
- **Section 6.3.7 -** LGN Horizontal Link Hello Protocol and PTSEs
- **Section 6.3.9 -** Aggregation Tokens and Link Aggregation

Because the three subsets mentioned above overlap in terms of the capabilities that they provide, each capability is described once in the following subsections.

6.3.1 Outside Links and Hello Protocol over Outside Links

In *ForeThought* 5.x, only four of the possible states of the PNNI Hello Finite State Machine (FSM) are implemented. These states include: down, attempt, one-way inside and two-way inside. In *ForeThought* 6.x, the FSM states that pertain to outside link support are implemented. These states include: one-way outside, two-way outside, and common outside.

An *outside link* is a link to a lowest-level outside node. It is not part of the PNNI topology like a horizontal link or uplink is, so it is not used in path computation. However, outside link support is essential for the PNNI LGN hierarchy because uplinks and SVCC RCCs are based on this support. Outside link support means that two PNNI nodes that belong to two different peer groups can be connected directly with each other by an outside link. When the two PNNI nodes realize this is an outside link, the corresponding Hello FSMs eventually change to the common outside state, as long as the two nodes share a common peer group ID at a higher level of the PNNI hierarchy. Otherwise, they change to the two-way outside state.

6.3.2 Origination of Uplinks and Uplink PTSEs

An *uplink* connects a border node to an upnode. A *border node* is node in a peer group that has at least one link which crosses the peer group boundary. An *upnode* is a node that represents a border node’s outside neighbor in the common peer group. The upnode must be a neighboring peer of one of the border node’s ancestors.

Each physical link that is in the common outside Hello state has a logical uplink irrespective of the aggregation token associated with the physical link. The origination of uplink PTSEs is fully implemented as defined in Section 5.10.2 of the ATMF PNNI Specification. An uplink PTSE is generated as soon as the link reaches the common outside Hello state.

6.3.3 Entry and Exit Border Node DTL Handling

ForeThought 6.x implements the handling of DTLs by entry and exit border nodes. An *entry border node* is the first node in a peer group which receives a call over an outside link. An *exit border node* is the last node in a peer group to see a call. It sends the call out of the peer group over an outside link. When a signalling call enters a peer group in a hierarchical PNNI network, the entry border node may need to add several DTLs to the beginning of the list of existing DTLs in order to process the call. Additionally, an exit border node may need to remove some DTLs from the list of existing DTLs when it forwards the call to the next peer group.

6.3.4 Entry Border Node Crankback Handling

ForeThought 6.x implements crankback capabilities for entry border nodes. Because an entry border node can generate DTLs, the entry border node may have to crank back when a link or port is blocked in a DTL that it originated.

6.3.5 SVCC RCC Establishment

When a PGL receives an uplink advertisement of an uplink from one of the peer group's border nodes, that advertisement contains information about the neighboring LGN's (Upnode's) node ID, peer group ID, ATM address, etc. The PGL uses this information to establish an SVCC RCC to the neighboring LGN. The SVCC RCCs are established as defined in Section 5.5.6.1 of the ATMF PNNI Specification.

6.3.6 SVCC RCC Hello Protocol

As soon as an LGN detects an uplink to a neighboring LGN or to a PNNI node (in a lop-sided hierarchy), it tries to establish an SVCC RCC so that it can communicate with the other LGN or PNNI node. Once the SVCC RCC is established, the LGN starts the SVCC-based RCC Hello Protocol as defined in Section 5.6.3.1 of the ATMF PNNI Specification. When the Hello state reaches the twoWayInside state, it triggers an `AddPort` event in the neighboring peer state machine. When this occurs, database summary packets, PTSE Req packets, PTSPs, and PTSE acknowledgment packets are exchanged.

6.3.7 LGN Horizontal Link Hello Protocol and PTSEs

The states of all the horizontal links to an LGN neighbor are determined from the information in a single LGN Horizontal Link Extension information group included in the Hellos that are sent over the SVCC-based RCC. Each distinct aggregation token value represents a distinct horizontal link with its own independent state machine. This aggregation token is present in the LGN Horizontal Link Extension IG. An LGN horizontal link is advertised in PTSEs originated by the LGN, if and only if, the LGN horizontal link hello state is twoWay. The LGN Horizontal Link Hello Protocol is fully implemented as described in Sections 5.6.3 and 5.6.3.2 of the ATMF PNNI Specification.

6.3.8 PGLs

6.3.8.1 Parent Nodes and Child Nodes

PGL capability is fully implemented as described in Sections 5.10.1 through 5.10.1.5 of the ATMF PNNI Specification. A *PGL* is a node in a peer group that collects and assimilates data to represent the entire peer group as a single node in the form of a parent node. A *parent node* is an LGN that represents its peer group at the next higher level of hierarchy. A *child node* is a node at the next lower level of hierarchy. It can be an LGN or a physical node.

6.3.8.2 PGL Election

The PGL is elected by the members in the peer group. The node with the highest leadership priority is chosen as PGL. After a node becomes PGL, its leadership priority is increased by 50 to ensure stability. The elections take place on a continuing basis such that when a node with a higher leadership priority than the current PGL becomes active, another election is held to transfer the leadership. Elections are also held when the current PGL fails or is deleted, so that the leadership is transferred to the node with the next highest priority. If more than one node has the same priority, the node with the highest node ID is declared PGL and its leadership priority is increased by 50 to ensure stability.

6.3.8.3 PGL Capability

The PGL performs the following functions:

- The PGL provides support for the `AwaitUnanimity`, `OperPGL`, and `HungElection` states of the PGL FSM. All of the other FSM states are already implemented in *ForeThought* 5.x.
- The PGL allows the PGL priority to be set to non-zero values.
- The PGL allows the creation of parent-child relationships between nodes to facilitate LGN activation.
- The PGL provides support for the activation of a parent LGN when a child node enters the `OperPGL` state in its PGL FSM.

6.3.9 Aggregation Tokens and Link Aggregation

6.3.9.1 Aggregation Tokens

An *aggregation token* is a number assigned to an outside link by the border nodes at the ends of the outside link. The same number is assigned to the uplink associated with the outside link. In all parent peer groups and higher level peer groups, all uplinks that have the same aggregation token are combined, or aggregated, and represented as one link.

Aggregation tokens are fully implemented as described in Sections 5.10.1 through 5.10.1.5 of the ATMF PNNI Specification. All links between a pair of LGNs with same aggregation token must be advertised as one logical link. Derived uplinks and LGN horizontal links are aggregated depending on the aggregation token values included in the PTSEs describing uplinks and horizontal links as binding information.

6.3.9.2 Derived Aggregation Tokens

If neighbors exchange configured aggregation tokens of the same value, then the derived aggregation token is that value (e.g., if both advertise 8, the derived value is 8; or if both advertise 0, the derived value is 0).

If one neighbor has a configured aggregation token of 0 and the other has a non-zero configured aggregation token, then the derived aggregation token is the non-zero value (e.g., if one advertises 4 and the other advertises 0, then the derived value is 4).

If neighbors exchange configured aggregated tokens of different non-zero values, then the derived aggregation token value is 0 (e.g., if one advertises 5 and the other advertises 6, then the derived value is 0).

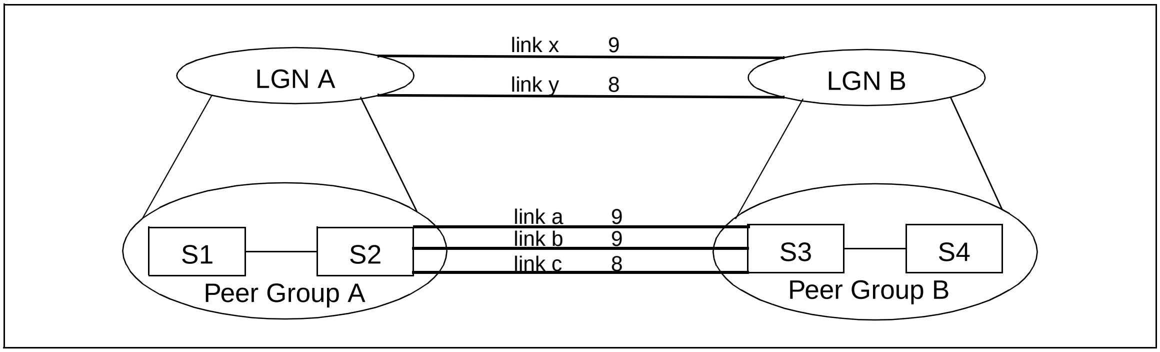

Figure 6.4 - Aggregation Example

In the aggregation example in Figure 6.4, switches S1 and S2 form peer group A and switches S3 and S4 form peer group B. Outside links a and b each have an aggregation token of 9, and outside link c has has an aggregation token of 8. Switches S2 and S3, which are border nodes, advertise three separate uplinks. However, links a and b are aggregated together as one LGN horizontal link, which is link x. There is another separate LGN horizontal link for link c, which is represented as link y.

6.3.9.3 Link Aggregation

Link aggregation is fully implemented as described in Section 5.10.3.1 of the ATMF PNNI Specification. *Link aggregation* is the process of representing several parallel links as a single higher-level link. The topology state parameters for this type of link are derived from those for the individual links being aggregated.

For example, suppose you have two outside links with same aggregation token as follows:

- link1 (OC-12c on an LC network module) Non-VP capable
- link2 (OC-3c on a Series D network module) VP capable

The LGN horizontal link advertises the RAIGs of the link with highest bandwidth (available cell rate) per service category and advertises it as VP capable. Even if only one of the aggregated links is VP capable, then the aggregated link is considered to be VP capable.

Therefore, in the example links given, the aggregated link advertises OC-12 values for all service categories, except for ABR, because ABR is not supported on LC network modules. The OC-3's values are advertised, instead, for ABR.

6.3.10 Nodal Aggregation

Nodal aggregation is implemented as described in Sections 5.10.3.2 and 5.10.3.3 of the ATMF PNNI Specification, except for bypasses and exceptions. Nodal state parameter information group is used to advertise elements of this node's complex node representation. All metrics and attributes for each input and output port must be included in the same IG. Additionally, the flag in the nodal IG is set for complex nodal representation as specified in Section 5.14.9.1.2 of the ATMF PNNI Specification.

6.3.11 Internal and Exterior Address Summarization

Address summarization is fully implemented as described in Section 5.9 of the ATMF PNNI Specification. When summarizing exterior reachable addresses, the aggregation of RAIGs associated with the address being summarized is performed only if there is a summarizing policy on that prefix with metrics (RAIGS) specified. Otherwise, no RAIG aggregation is performed.

6.3.12 Anycast Address Scoping

Scope defines to what level in the PNNI routing hierarchy an address should be advertised. (See Section 6.4.3.3.2 in this manual for more information about scope.) Anycast address scoping is implemented as described in Section 7.0, Annex 5, and Annex 6 in the UNI 4.0 Specification; Section 9.4.2.4 in the ILMI 4.0 Specification; and in Sections 3.5.1, 5.3.6, and 5.9.1, and Appendix A in the PNNI v1.0 Specification.

The handling of address scoping for anycast (group) addresses consists of two parts:

- the mapping of organizational scopes into routing levels
- the handling of the connection scope selection IE during connection establishment

6.3.12.1 Organizational Scope and PNNI Scope

The organizational scope of an anycast address is learned by the switch when the end device registers the anycast address through ILMI registration. The organizational scope is mapped into a routing level following a mapping table that is configured using the pnniScopeMappingTable table in the PNNI MIB. The mapping occurs when the address is registered as reachable through a PNNI node. Then this information is stored in Prefix Table Entries (PTEs) and flooded to other nodes through PTSEs.

The PNNI scope mapping levels can be configured for each PNNI node through SNMP. If no levels are configured, then default values are used. Similarly at ILMI registration, if no scope is specified, then default values are used. The default values for organizational scope are localNetwork for ATM group addresses, and global for individual addresses.

6.3.12.2 Connection Scope Selection

The connection scope selection IE also gets mapped into a PNNI routing level. The mapping takes place at the DTL originator. The PNNI routing level (PNNI node’s level) is used to select PTEs that satisfy the scoping requirement. Only those that have been registered with the originator’s level that are numerically greater or equal to the mapped connection scope selection are eligible for path selection. At the entry border node to the destination peer group, further checking is performed. This time, the connection scope selection is not used. The scope of the highest level DTL is used instead, and is compared to the advertisement scope of the PTEs.

There are two types of ATM endsystem addresses: individual and group addresses. An ATM group has one or more members, and an ATM end system can be a member of zero or more ATM groups at any time. The purpose of the connection selection scope IE is to allow the calling user to indicate to the network the selection range/domain. A setup message with ATM group address is received by the network. If the network does not find any registered addresses or members with the indicated connection scope, then the network releases the call with cause 3 (no route to destination). For ATM group addresses, if no connection selection scope is specified in the setup, then routing uses localNetwork as the default to route to the group address. For PNNI routing, unicast or anycast/group addresses are treated alike.

6.4 Internetworking between PNNI and FT-PNNI

Some networks running FT-PNNI may be connected to networks running PNNI. To provide internetworking of these protocols, *ForeThought* 5.x and 6.x allows reachability information to be leaked dynamically between FT-PNNI and PNNI peer groups.

6.4.1 Gateway Switches and Split Switches

6.4.1.1 Using *ForeThought* 5.x

When you upgrade a switch to *ForeThought* 5.x, it has a single FT-PNNI node by default. If you change the default domain protocol to gateway, a second node is created automatically on the same switch that runs the PNNI protocol, so you have both a single FT-PNNI node and a single PNNI node running concurrently within a single switch. This is called a *gateway switch.* A gateway switch connects a PNNI network with a FT-PNNI network.

Using *ForeThought* 5.x, you can also create a second PNNI node on single switch. A switch that is running two or more PNNI nodes is called a *split switch.* A split switch connects two or more PNNI areas.

The FT-PNNI and PNNI nodes within a gateway switch and the multiple nodes within a split switch do not share topology information, but they dynamically exchange reachability information to facilitate connectivity between a source in one network and a destination in another. The process for this exchange of reachability information is discussed in Section 6.4.3.

As far as DTL processing is concerned, a gateway switch or a split switch appears as either a DTL terminating or DTL originating node. For example, Figure 6.5 shows that S3 is a DTL terminating node for the FT-PNNI DTL and a DTL originating node for the PNNI DTL for a connection going from switch S1 to switch S4.

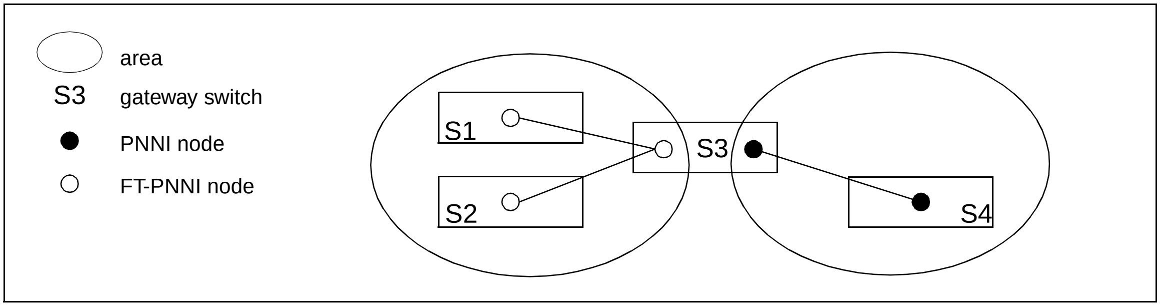

Figure 6.5 - Internetworking of FT-PNNI and PNNI

ATM Forum PNNI

6.4.1.2 Using *ForeThought* 6.x

When you upgrade a switch to *ForeThought* 6.x, it has a single PNNI node by default. Using *ForeThought* 6.x, you can have up to five levels of hierarchy using PNNI. You can create a split switch with a maximum of five lowest level nodes. All five nodes can be at the same forelevel or four can be at the same forelevel with one at a higher (numerically smaller) forelevel. Each of these nodes can have a PNNI instance, and each PNNI instance can have one lowest level node and up to nine LGNs.

You can also create a gateway switch with a maximum of four lowest level PNNI nodes and one lowest level FT-PNNI node. All five nodes can be at the same forelevel or four can be at the same forelevel with one at a higher (numerically smaller) forelevel.

6.4.2 Interfaces

Interfaces can be attached to PNNI nodes. By default, a PNNI interface comes up attached to the default PNNI node in *ForeThought* 6.x. FT-PNNI interfaces are automatically attached to FT-PNNI nodes. An interface becomes a PNNI or FT-PNNI interface via ILMI auto-configuration or by the user hard configuring the UNI using the command **conf sig new**. (For more information about this command, see Part 2 of the *AMI Configuration Commands Reference Manual*.)

6.4.3 Dynamic Leaking of Reachability Information

This feature allows dynamic leaking of reachability information between the single FT-PNNI node and the single PNNI node within a gateway switch. It may also be used to control the leaking of reachability information between multiple PNNI peer groups. Using split switches, it is possible to connect and exchange reachability information between any number of PNNI peer groups. This dynamic leaking is controlled using the following user-configurable settings: grouping of nodes within a switch into areas and domains, and configuring policies that allow you to summarize, suppress, or advertise reachability addresses between peer groups. See Section 6.4.3.3.1 for more information about policies. Nodes within a switch that belong to different domains do not exchange information.

6.4.3.1 Areas

An *area* is a subset of nodes within a domain that are contiguous and that together execute a link state routing protocol to exchange reachability information dynamically among themselves. Because of the database exchange protocol, two nodes that belong to an area have identical copies of the link-state topology database. Using *ForeThought* 5.x, an area can be one of the following:

- a non-hierarchical PNNI network (i.e., a single PNNI peer group because only the lowest-level peer group is implemented)
- a hierarchical FT-PNNI network (i.e., a multiple FT-PNNI peer group system using hierarchy)
- a non-hierarchical FT-PNNI network (i.e., a single FT-PNNI peer group)

Using *ForeThought* 6.x, an area can be any of the above or can be the following:

- a hierarchical PNNI network (i.e., a multiple PNNI peer group system using hierarchy)

Split switches can be used to connect multiple PNNI areas and gateway switches can be used to connect an FT-PNNI area and a PNNI area, as shown in Figure 6.6. The only restriction in configuring areas is that a FT-PNNI area cannot adjoin another FT-PNNI area.

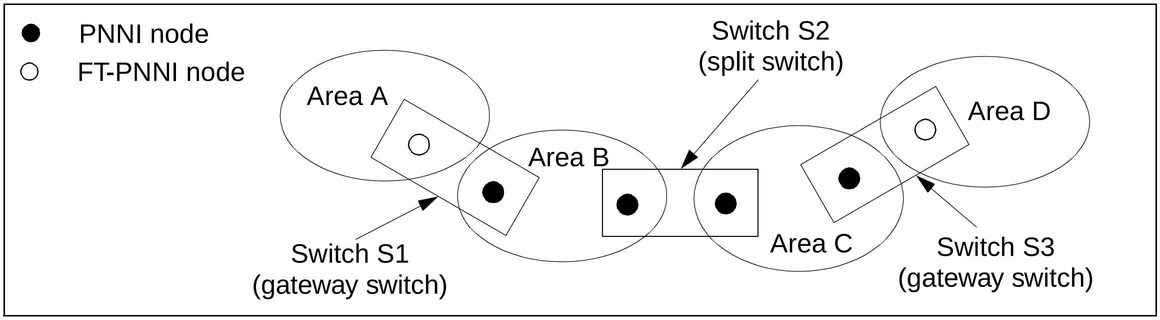

Figure 6.6 - Split Switches and Gateway Switches Connecting Areas

6.4.3.1.1 Peer Groups in Areas

Multiple peer groups can exist within an area. Peer groups within an area are connected to each other by border links as shown in Figure 6.7.

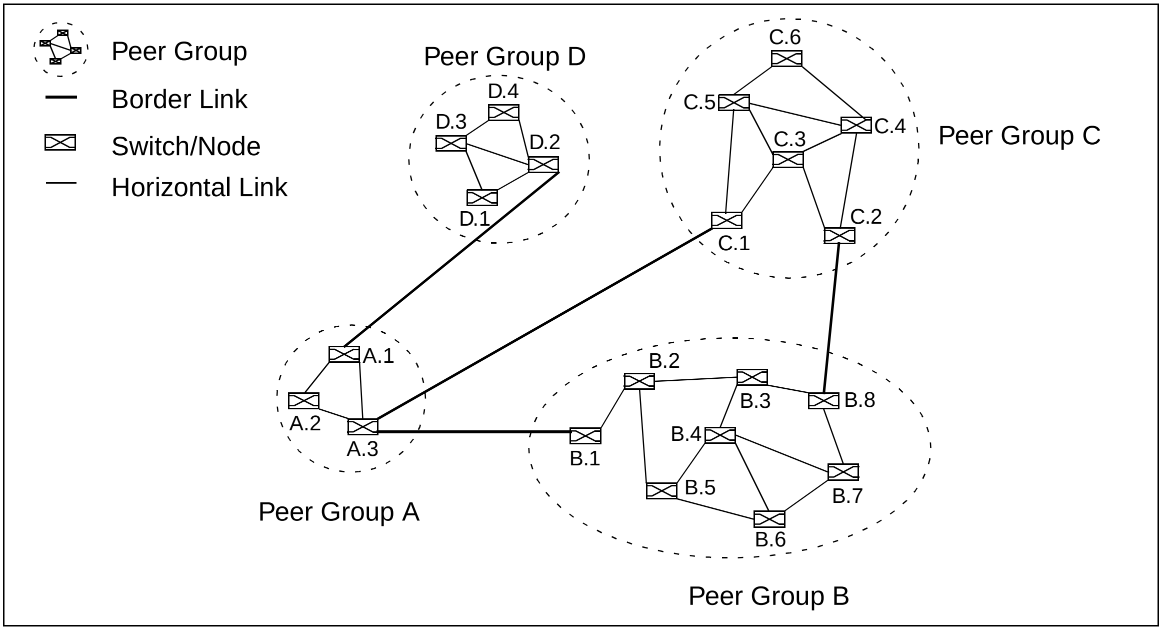

Figure 6.7 - Peer Groups Connected by Border Links

6.4.3.1.2 Area IDs

Each area has an area ID which is unique only within a domain. The area ID is only useful at the border of the area. Since area IDs are included in some of the routing packets distributed throughout a domain, all switches belonging to the same area must be configured with the same area ID. It is especially important for the nodes in split switches and gateway switches to be configured with the correct area ID.

6.4.3.1.3 Levels

Each area has a *level* associated with it. Levels are used to control the flow of reachability information in a multi-level routing hierarchy (which is discussed in the next section). An N-level hierarchy has N discrete values of levels associated with it, namely, L_1, L_2,..., L_N. Each of the areas in the hierarchy has one of these N values as its level. Numerical values are assigned to levels in order from the lowest to the highest; i.e., $L_1 < L_2 < L_3 < L_{N,}$ so L_N is the highest level. An area of level L_i can adjoin any number of areas at level L_{i-1}, but can only adjoin at most one area at level L_{i+1} or higher. This means that on a given switch, there can never be areas configured that belong to more than two distinct levels. Furthermore, there can only be one area that belongs to the higher level.

6.4.3.2 Domains

A *domain* is a group of areas that are configured to dynamically exchange reachability information with one another. This allows connectivity between end systems belonging to different areas without configuring static routes *between areas*. However, reachability information is exchanged *between domains* only if inter-domain routes are manually configured between the two domains. Figure 6.8 shows an example of how an inter-domain route can connect two domains.

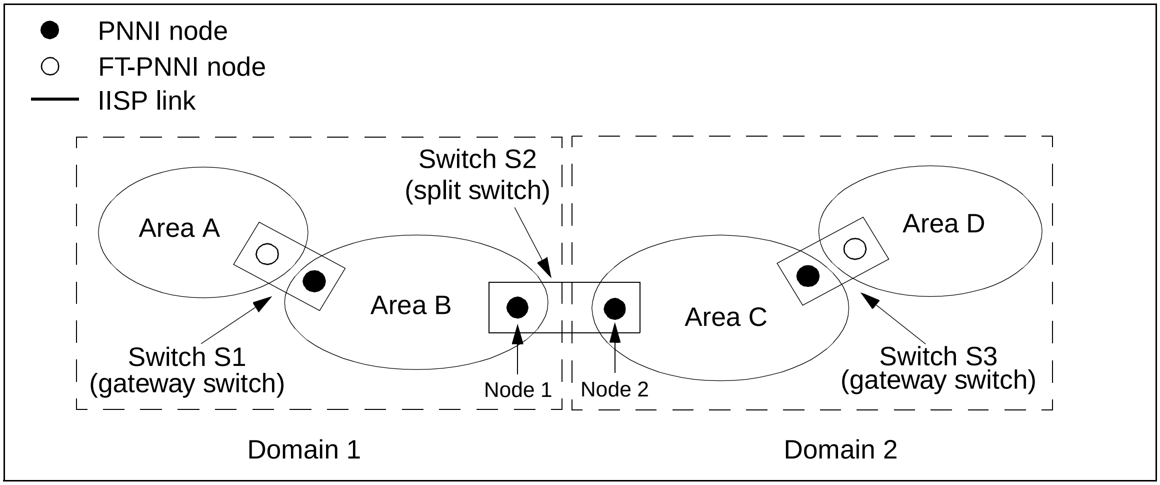

Figure 6.8 - Inter-domain Route Connecting Two Domains

In Figure 6.8, the two nodes configured in switch S1 belong to different areas, and so they do not share the same link-state topology database. The same is true for the nodes configured in switch S3. Dynamic reachability leaking takes place among the nodes in switch S1, enabling end systems in Area A to reach other end systems in Area B, and vice versa within Domain 1. The same applies to the nodes in switch S3.

The two nodes in split switch S2 belong to two different areas, each of which is part of a different domain. An inter-domain route must be configured on both node 1 and node 2 so they can share reachability information between two domains. An inter-domain route can be created using the command **`conf atmr pnni address new interdomain`**. Exchange of reachability information can then occur between the nodes over this inter-domain route.

6.4.3.2.1 Configuring Domains

Domains only need to be configured on split switches and gateway switches; i.e., switches that belong to multiple areas. Other switches operate properly without domain configuration.

On switches running *ForeThought* 5.x, one FT-PNNI domain (with domain index 1) is pre-configured by default. On switches running *ForeThought* 6.x, one PNNI domain (with domain index 1) is pre-configured by default. The default domain cannot be deleted, but the information within the domain can be changed. Multiple domains can be configured on a switch. For each domain, the following parameters can be changed:

- The domain name, which is optional, can be configured for easier manageability. However, the name of the default domain cannot be changed.
- The domain protocol, which sets the routing protocol for the switch, can be set to `ftpnni`, `pnni`, or `gateway`. This parameter can also be changed on the default domain. On switches running *ForeThought* 5.x, the default setting is `ftpnni`, so if you want to configure a split switch, you must change the domain protocol to `pnni`. On switches running *ForeThought* 6.x, the default setting is `pnni`, so if you want to configure a split switch, you simply need to create one or more additional PNNI nodes on the switch. On switches running *ForeThought* 5.x or 6.x, if you want to configure a gateway switch, you must change the domain protocol to `gateway`.
- The domain prefix can be changed. This sets the default 13-byte NSAP prefix used by the switch’s routing protocol(s).
- A signalling interface (port/VPI) can be associated with a particular domain, while the routing interface (port/VPI) for that signalling interface can be associated with a particular routing node within that domain.

For more information about changing these parameters, see Chapter 4 in Part 1 of the *AMI Configuration Commands Reference Manual.*

6.4.3.3 Propagation of Reachability Information

The leaking of reachability information between two areas is constrained by two things: policy and scope.

6.4.3.3.1 Policies

ForeThought software lets you configure a policy as a flexible means of enforcing security across the network topology. When a node discovers an address that is leaked by another node, it checks its own *policy* table to determine if the address is to be advertised within its own area. There are three types of policies:

- summary - Addresses matching a `summary` policy cause just the summarized prefix to be announced to the node's peer group instead of advertising each prefix individually.
- suppress - Addresses matching a `suppress` policy are not announced to the node's peer group at all.
- advertise - Addresses matching an `advertise` policy cause the entire address to be announced to the node's peer group.

6.4.3.3.1.1 Policies in a Flat PNNI Network

In each node's policy table, the switch chooses the best address match, meaning the longest prefix match, to determine which policy applies to an address. Therefore, the `suppress` and `advertise` policies provide limited filtering since addresses matching a broad prefix can be filtered at the split switch or gateway switch through the `suppress` policy, then a particular service, device, or switch can be exempted by advertising an address within the suppressed range. The `summary` policy is best for scalability since one address prefix is shared, rather than dozens or hundreds of more specific addresses.

For example, suppose you have an area A which contains peer groups A.1 and A.2. Suppose you create two policies: one says suppress all address that are area A and the other policy says advertise all address that are peer group A.1. If an address A.1 comes in to the area, the policy for advertise takes precedence over the policy for suppress (the advertise policy is a longer prefix match). Therefore, the address is advertised. However, if an address A.2 comes in to the area, it is suppressed (since there is no advertise or summary match for A.2).

6.4.3.3.1.2 Policies in a Hierarchical PNNI Network

In an LGN hierarchy, when you assign a policy to an LGN, the information that is present in the lower level peer group, or in the contained peer group, gets summarized, suppressed, or advertised, depending on the type of policy configured. For example, suppose you have a peer group B.1.1 which has a higher level LGN B.1. If there are exterior reachable addresses in peer group B.1.1, you can apply a policy to LGN B.1 which suppresses or summarizes those addresses.

6.4.3.3.2 Scope

6.4.3.3.2.1 Scope in a Flat PNNI Network

Each piece of reachability information has a *source area ID* and a *scope* associated with it. The source area ID of reachability information originated in a given area is 0 within that area, meaning that is local. The source area ID of reachability information advertised from, for example, area A1 into area A2 has source area ID A1.

When a leaked address is to be advertised or summarized, a node also determines the scope with which to advertise the address (or its summary). The scope denotes the highest level in the PNNI routing hierarchy at which an address can be advertised. Scoping ensures that reachability information does not loop. A reachable address being advertised with a scope of l can only be leaked to areas of level lower (larger in numerical value) than l. The scope of reachability information originated in a given area is 0 within that area, meaning that it is local. The scope of reachability information advertised from, for example, area A1 (at level l_1) into area A2 (at level l_2) is l_2 if l_2 is a lower level than l_1, but is l_1 if l_2 is a higher level than l_1.

6.4.3.3.2.2 Scope in a Hierarchical PNNI Network

See Section 6.3.12 in this manual for information about scope in a hierarchical PNNI network.

6.4.3.3.3 The Process for Leaking Reachability Information

The following process is used by a node to determine if and how a leaked reachable address is to be advertised to its peer group:

1. The node finds the longest matching policy prefix for the address.
2. If there is no matching policy prefix, the leaked address is advertised as is. If there are multiple addresses with different scopes, then multiple addresses are advertised.
3. If the longest matching policy prefix is a policy suppress, the leaked address is not advertised at all.
4. If the longest matching policy prefix is a policy advertise or summary, the policy prefix is advertised or summarized.
5. If there are multiple leaked addresses with different scopes for which the policy prefix is longest matching, then the one with the widest scope is used to determine the scope with which to advertise or summarize the policy prefix.

The following rule determines the scope with which to advertise an address or a policy prefix:

- If the address used to determine the scope (chosen above) was leaked from a strictly lower level (numerically larger) area, then the scope of the address is used; otherwise, the level of the area to which the advertising node belongs is used.

Metrics can optionally be configured with a policy prefix. In this case, the configured metrics are included in the advertisement of the policy prefix.

6.5 Path Extension QoS

This feature allows QoS parameters (CTD, CDV, and CLR) to be configured for VPs. For a PNNI interface that is configured on a VPs, the associated QoS parameters are computed as the combination of the parameters configured on the VP and those that are part of the switch and network modules.

This feature also allows a particular service category to be configured as "none," in which case the associated PNNI link does not advertise the service category, thus preventing connections with the given service category from using the link. For more information about assigning a path extension to a VP, see Chapter 20 in Part 2 of the *AMI Configuration Commands Reference Manual*.

6.6 Load-Balanced Routing

As a value-added feature, you can perform load-balanced routing on FORE switches when using ATMF PNNI. This feature lets you configure a node to distribute CBR, VBR, ABR, or UBR traffic more evenly across available links.

ATMF PNNI path computation is done in two steps. In the first step, the shortest path is computed using only links that satisfy the profile's QOS requirements. For a load-balanced profile, only one shortest path tree is computed, but up to four alternate links that lead to the equal cost shortest paths are stored at each hop to be used in the second step. For a non load-balanced profile, three shortest paths are computed and optimized for administrative weight, CTD, and CDV.

In the second step, a DTL is constructed by tracing the path from the destination to the source. For load-balanced profiles, up to four choices are available at each hop and one is selected randomly, thereby distributing the calls across the network. However, because the links are selected randomly, there is no guarantee of satisfying the connection's CTD and CDV requirements. Therefore, when a load-balanced profile fails to meet the CTD and CDV requirements, a non load-balancing profile is used to compute the path. If load-balancing is enabled on a node, then any user-defined profile with load-balancing disabled is ignored. Similarly, if load-balancing is disabled on a node, then any user-defined profile with load-balancing enabled is ignored.

This feature is enabled (set to true) by default on a per-node basis, but it can be disabled (set to false) during the creation of a node by entering **`conf atmr pnni node new`** *`<index>`* **`-loadbal false`**. It can also be enabled or disabled on an existing node using **`conf atmr pnni node modify`** *`<index>`* **`-loadbal`** (**`true`** | **`false`**). For more information about these commands, see Chapter 4 in Part 1 of the *AMI Configuration Commands Reference Manual*.

6.7 Configuring PNNI

This section assumes that you are configuring PNNI for the first time. If you already have an FT-PNNI network that you want to convert to PNNI, or if you already have a flat PNNI network running and you want to change it to a hierarchical network, please see Appendix C in this manual for examples of how to convert those networks.

Before you configure a PNNI network, it is assumed that you are familiar with the fundamentals of PNNI routing and that you are familiar with FORE's implementation of PNNI as described in the earlier sections of this chapter.

There are a few major steps that the system administrator should follow in order to configure a PNNI network.

1. Do preparation work first to determine the following:
   a. Decide how to divide the NSAP address space.
   b. Decide how many layers of hierarchy you need.
2. Configure the NSAP addresses on all of the switches.
3. Choose a PGL.
4. Configure the parent nodes.
5. Configure the PGL priority.
6. Configure the remaining peer groups.
7. Verify that the hierarchy has come up.

The remainder of this section gives an example of configuring a PNNI network.

To configure hierarchical PNNI on a switch, you will log into AMI on a switch running *ForeThought* 6.0.x or greater and use the commands found under **`configuration atmr`**.

NOTE

More information about each of these commands may be found in Part 1 of the *AMI Configuration Commands Reference Manual.*

6.7.1 Doing the Preparation Work

Before you configure any switches, you need to determine what you want your network to look like. It is recommended that you sketch the intended topology on paper first to determine the best way to segment the switches into peer groups.

6.7.1.1 Dividing the NSAP Address Space

Determining the NSAP addresses is an important task when configuring a PNNI network. You need to assign a common prefix to each member in a peer group. In this way, each peer group can be identified outside of the group as a single node by its address prefix. This conserves switch memory by reducing the size of the routing table.

There are three address formats: DCC, ICD, and E.164. FORE switches implement the ICD ATM format. Per the UNI 3.0 specification, all private networks should accept initial call setup messages containing ATM addresses with any of the approved formats and forward the calls as necessary.

An NSAP address consists of the following:

- a 13-byte network-side prefix - The prefix is the NSAP prefix of the switch to which the host is attached.
- a seven-byte user-side part - This consists of the following:
    - a six-byte End System Identifier (ESI) - The ESI is the unique IEEE MAC address of the interface.
    - a one-byte selector field

6.7.1.2 Address Summarization

A summarized address is represented by a full NSAP address plus a bit mask, which is referred to as a level. The level indicates the summary prefix for the address. That is, the level is the length of the bit mask at which the address prefix becomes significant to PNNI routing computations. This facilitates hierarchical routing because only summarized topology information is advertised outside of a peer group.

The 13-byte prefix consists of 104 bits. The peer group ID (PGID) is calculated as (104 - *n*) rightmost bits set to 0, where *n* is the level.

6.7.1.3 Determining the Layers of Hierarchy Needed

Address assignment to a node is determined by its location in the network topology. This allocation of bits should allow for topology growth both from the top and bottom of the hierarchy to ensure maximum scalability and flexibility. In Section 6.7.1.3.1, an enterprise design example is given. In Section 6.7.1.3.2, a service provider design example is given. In Section 6.7.1.3.3, a configuration example is given that is used throughout the rest of this chapter.

6.7.1.3.1 Enterprise Example

In a 13-byte prefix, the first 7 bytes are fixed. The next 6 bytes can be used for hierarchical addressing. The following is one example of how you can use your addressing space:

- Reserved - 1 byte
- Region - 1 byte
- Campus - 1 byte
- Building - 1 byte
- Closet - 1 byte
- Switch fabric - 1 byte

The following is an example of an NSAP prefix in ICD format:

Table 6.2 - Example Enterprise NSAP Prefix

Fixed: Specified by the Registration Authority							**Variable: Used for Hierarchical Addressing Scheme**					
Reserved							Rsvd	Region	Campus	Bldg	Closet	Fabric
1	2	3	4	5	6	7	8	9	10	11	12	13
47	00	05	80	FF	E1	00	F2	15	XX	XX	XX	XX

6.7.1.3.2 Service Provider Example

In a 13-byte prefix, the first 7 bytes are fixed. The next 6 bytes can be used for hierarchical addressing. The following is one example of how you can use your addressing space:

- Reserved - 1 byte
- Backbone - 1 byte allows for 256 BB nodes
- Region - 1 byte allows for 256 regions within the backbone
- Metropolitan area - 1 byte allows for 256 metropolitan areas within each region
- Customer - 2 bytes allows for 65,536 customers within each metropolitan area

The following is an example of an NSAP prefix in ICD format:

Table 6.3 - Example Service Provider NSAP Prefix

Fixed: Specified by the Registration Authority							**Variable: Used for Hierarchical Addressing Scheme**				
Reserved							Rsvd	BB	Region	Metro	Customer
1	2	3	4	5	6	7	8	9	10	11	12 and13
47	AB	CD	EF	12	34	56	XX	XX	XX	XX	XXXX

6.7.1.3.3 Configuration Example

The configuration example in Figure 6.9 shows a small network with 13 switches. The switches are going to be divided into three peer groups that are each at level 80 (the default for FORE switches). These three peer groups are going to be summarized by LGNs in one peer group at a higher level of 72. This is an example of a simple two-level hierarchy.

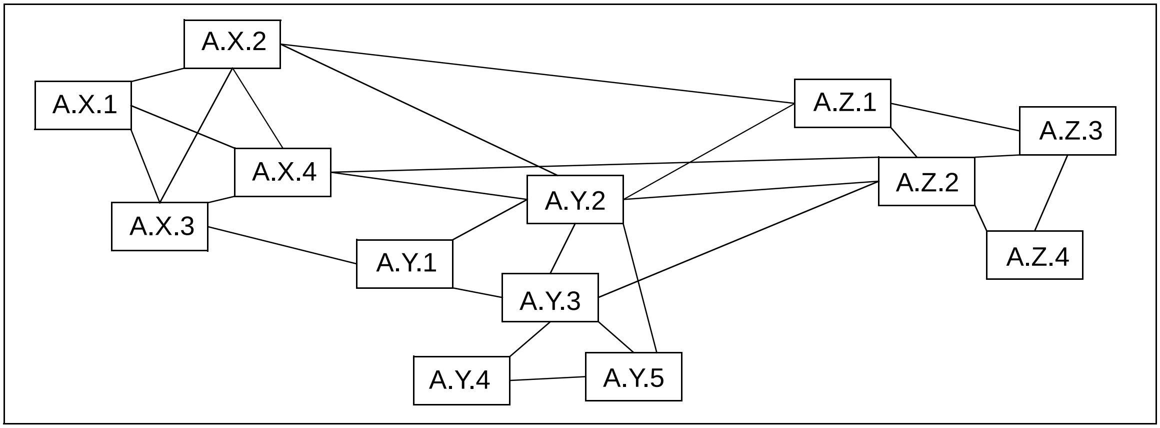

Figure 6.9 - Example Network to be Configured

6.7.2 Configuring the NSAP Addresses

Configure one peer group at a time. In the example shown in Figure 6.9, start with what will become peer group A.X and begin with switch A.X.1.

1. Display the default protocol for switch A.X.1. Be sure that all switches are running PNNI as shown in the `Defproto` field under **`conf atmr domain show`**.

```
switch_ax1::configuration atmroute> domain show
 ID  Name               Defproto   Defsum   Prefix
 1   default            pnni       DISABLED 0x47.0005.80.ffe100.0000.f31a.0172
                                     DefaultPrefix
                                     0x47.0005.80.ffe100.0000.f21a.26a3
```

PNNI is the default protocol on switches running *ForeThought* 6.0.x and greater. If any switches are running FT-PNNI, you need to change the default protocol to PNNI using the **`conf atmr domain modify`** command. You can change the protocol when you use this command to modify the prefix in step 3.

2. Check the connections between the switches that will become peer group A.X (in this case, 1B1 and 1D4) to be sure that the `Interface` field shows PNNI, the `SigVersion` is pnni10, and the `State` is up.

```
switch_ax1::configuration atmroute> conf sig show
 Port  VPI Interface      SigVersion State ILMI Side    RemoteAddress
 1B1     0 PNNI(a)        pnni10(a)  up    up   network 172.19.4.173
 1D1     0 privateUNI(a)  uni30(a)   down  down network
 1D2     0 privateUNI(a)  uni30(a)   down  down network
 1D3     0 privateUNI(a)  uni30(a)   down  down network
 1D4     0 PNNI(a)        pnni10(a)  up    up   network 172.19.13.219
 1CTL    0 privateUNI     uni30      up    down network
```

3. Modify the domain prefix of switch A.X.1 as follows:

NOTE

You checked the default protocol in step 1. If the default protocol currently is FT-PNNI, you should type in **`pnni`** after the domain ID when entering this command.

```
switch_ax1::configuration atmroute> domain modify <domain ID> [-name <name>]
[<pnni | ftpnni | gateway>][-prefix <prefix>] [-defsum <enable | disable>]
```

```
switch_ax1::configuration atmroute> domain mod 1 -prefix
47.0005.80.ffe100.0000.f200.0172

This operation requires the switch to be rebooted.
Any mismatch between the configured domain protocol and the interface type of
an existing UNI will result in that UNI not being created after reboot.
Reply ‘n’ to discard/correct any changes made so far.
Reboot the switch [n]? y
```

4. Repeat step 1 through step 3 on each switch in peer group A.X and assign the same prefix to the other switches in that peer group.

6.7.3 Choosing a PGL

You need to determine which switches should be PGL capable (eligible to be elected as PGL) for each peer group. When determining the best switches to be PGL, select switches with several links that are well connected to all of the other members of its peer group. It is best to choose two or three nodes to be PGL capable in case the PGL becomes disabled (e.g., a link gets broken). If the PGL goes down, another election is held, and a new PGL is elected.

NOTE When configuring a PGL at any level of hierarchy, it is highly recommended that you use a switch that has a 32MB SCP or greater to be the PGL.

6.7.4 Configuring the Parent Nodes

In the example in Figure 6.9, the best choices for PGL in peer group A.X are A.X.2 and A.X.4.

1. Configure a parent node only for the switches in peer group A.X that should have PGL capability. Begin with switch A.X.2 as follows:

```
switch_ax2::configuration atmroute> pnni node newparent <parentix> <childix>
[-level <level>] [-pglprio <pglprio>]
switch_ax2::configuration atmroute> pnni node newparent 2 1
```

   The default level is 80. When a parent node is created, the level is decremented by 8 by default. Do not assign a PGL priority yet. Create all of the parent nodes for the peer group and then set the PGL priority to a non-zero value on the PGL candidates.

2. Use the following command to display information about the parent node:

```
switch_ax2::configuration atmroute> pnni node show 2
 Node PnniNodeID                                          Level Lowest Rstrn
 2    72:80:47.000580ffe1000000f3000000.ff1a2e760002.00   72    false  false
      PeerGroupID                    PnniAtmAddress
     72:47.000580ffe100000000000000 47.000580ffe1000000f31a0519.ff1a2e760002.00
      Ptses OperStat AdminStat Shutdown ForeLevel ForeArea DomainId  DomainName
      0     down     up        false    5         5        1
      LoadBalance
      true
```

   The first part of the `PnniNodeID` is 72. This shows the level of this node (the parent node). The second part of the `PnniNodeID` is 80. This shows the level of the child node. The `OperStat` is down because the PGL has not been configured and elected yet.

6.7.5 Configuring the PGL Priority

In the example in Figure 6.9, the best choices for PGL in peer group A.X are A.X.2 and A.X.4. Switches A.X.2 and A.X.4 should be configured with a PGL priority (**-pglprio**) that is nonzero, which means they are eligible for election as PGL. The range of values is from 0 - 205. The other switches A.X.1 and A.X.3 already have the default PGL priority of 0. Leave their priority at 0 so they are not eligible to be elected.

1. Configure the PGL priority on the switches that you want to be eligible for election. In this example, they are switches A.X.2 and A.X.4.

```
switch_ax2::configuration atmroute> pnni node mod 1 -pglprio 100
switch_ax4::configuration atmroute> pnni node mod 1 -pglprio 50
```

2. The election process may take a minute or two. Display the PGL information for the nodes on switch A.X.2 to see if the election is complete. If the `PglState` says `operPgl` when the node has been elected as PGL.

```
switch_ax2::configuration atmroute> pnni node show pgl
 Node PglPrio(Cfg) AdvPglPrio ParentIx PglInitTime PglOverDelay PglReelect
                                       (sec)       (sec)        (sec)
 1    50           100        2        15          30           15
   PglState        PreferredPgl
   operPgl         80:160:47.000580ffe1000000f31a0519.ff1a2e760001.00
   ActiveParentNodeId
   72:80:47.000580ffe1000000f3000000.ff1a2e760002.00
   CurrentPgl                                          PglTimeStamp
   80:160:47.000580ffe1000000f31a0519.ff1a2e760001.00  TUE JAN 26 19:24:28 1999

 Node PglPrio(Cfg) AdvPglPrio ParentIx PglInitTime PglOverDelay PglReelect
                                       (sec)       (sec)        (sec)
 2    0            0          0        15          30           15
   PglState        PreferredPgl
   starting        0:0:00.000000000000000000000000.000000000000.00
   ActiveParentNodeId
   0:0:00.000000000000000000000000.000000000000.00
   CurrentPgl                                          PglTimeStamp
   0:0:00.000000000000000000000000.000000000000.00     N/A
```

NOTE

The advertised PGL priority (`AdvPglPrio`) shows the user-configured priority for all nodes except the PGL. Because the priority of the PGL is incremented by 50 to ensure stability, the `AdvPglPrio` of the PGL shows the user-configured priority plus 50.

6.7.6 Configuring the Remaining Peer Groups

After the PGL has been elected for peer group A.X, the network looks like Figure 6.10 with LGN A.X representing peer group A.X. LGN A.X is at level 72, while the rest of the lower level nodes are at level 80.

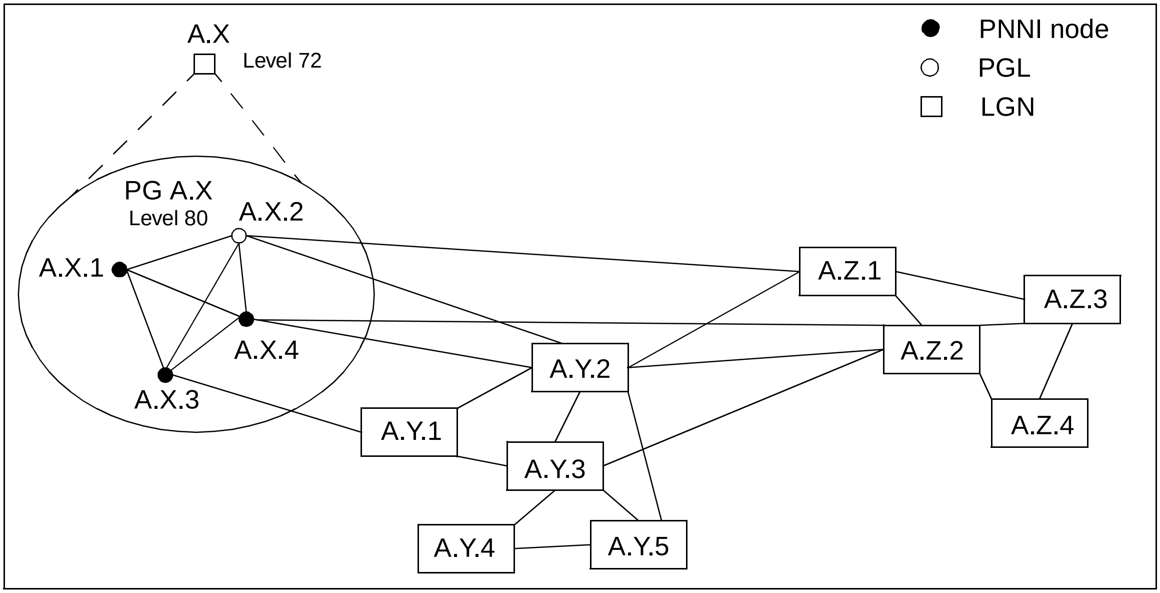

Figure 6.10 - LGN A.X Represents Peer Group A.X at Level 72

Repeat the steps in Section 6.7.2 through Section 6.7.5 to configure peer groups A.Y and A.Z. Once you have configured peer groups A.Y and A.Z, the network looks like Figure 6.11. LGN A.Y is at level 72 representing peer group A.Y, while the rest of the lower level nodes are at level 80. Similarly, LGN A.Z is at level 72 representing peer group A.Z, while the rest of the lower level nodes are at level 80.

The LGNs form peer group A at level 72. The LGNs dynamically establish SVCC RCCs to each other and begin to communicate.

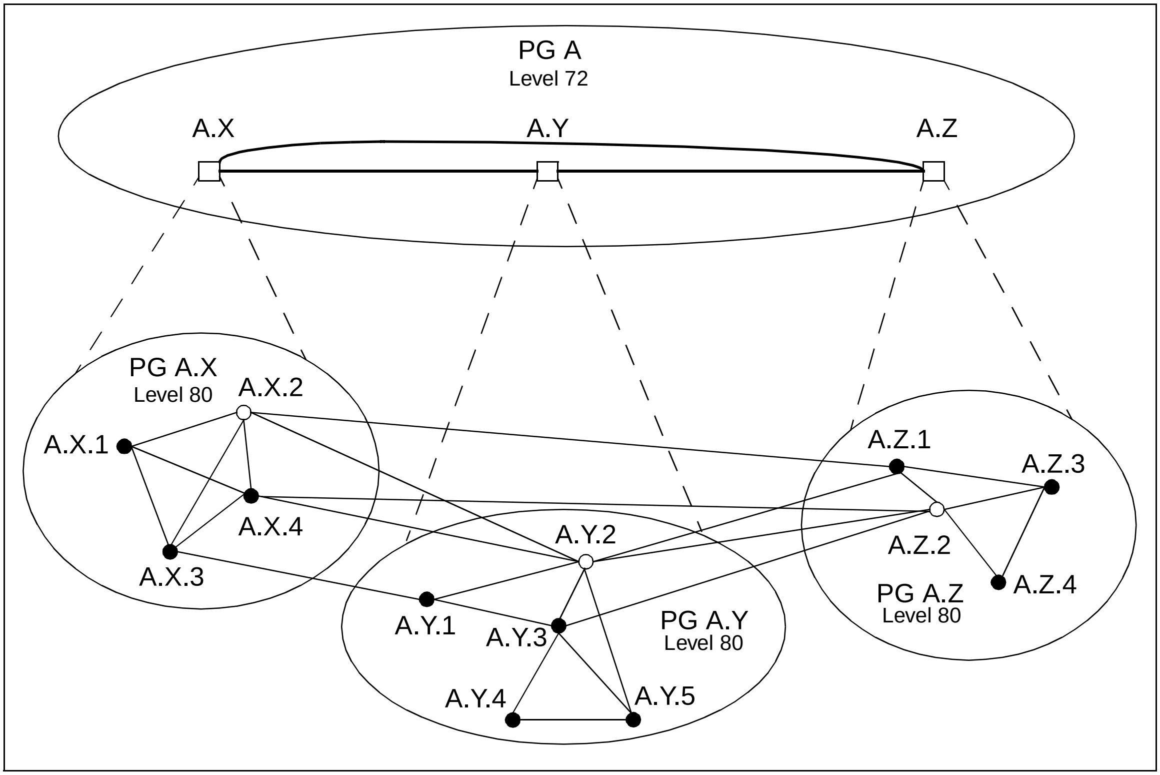

Figure 6.11 - Completed Peer Groups

6.7.7 Verifying That the Hierarchy Has Come Up

There are several things you can check to ensure that the hierarchy is working properly.

1. Check the `HelloSt` on the outside link between any two peer groups as follows:

```
switchaz2::configuration atmroute> show
 Port VPI  Node   Domain SigProto       SigSt  NodeSt HelloSt       PeerSt
 1A1  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1A2  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1A3  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1A4  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1B1  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1B2  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1B3  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1B4  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1C1  0    1      1      PNNI(a)        up     up     twoWayInside  full
 1C2  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1C3  0    1      1      PNNI(a)        up     up     commonOutside N/A
 1C4  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1D1  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1D2  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1D3  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1D4  0    N/A    1      privateUNI(a)  down   N/A    N/A           N/A
 1CTL 0    N/A    1      privateUNI     up     N/A    N/A           N/A
```

When everything is configured properly, the `HelloSt` on the outside link between two peer groups (border link) goes to `commonOutside` as shown on port 1C3. If the link still shows `twoWayOutside`, then check the following items:

a. Ensure the prefixes have been properly configured using **`conf atmr domain show`**. Remember that each switch in a peer groups shares a prefix, and each peer group must have a different prefix than the other peer groups.

b. Ensure the parent nodes have been properly configured and that their `OperStat` is `up` using **`conf atmr pnni node show`**.

c. Ensure that the PGLs have been properly configured and that their `PglState` is `operPgl` using **`conf atmr pnni node show pgl`**. The border link still shows `twoWayOutside` if the neighboring peer group does not have a PGL configured yet or if the neighboring peer group's PGL has not come up yet.

2. Check that an SVCC RCC has been established between LGNs by using the following command on each LGN. If the SVCC RCC has been established properly, the `HelloState` shows `twoWayInside`.

```
switchay2::configuration atmroute> display atmr pnni svccrcc
 Node SvccRccIx Version        HelloState    RcvHellos XmtHellos
 2    0         version1point0 twoWayInside  15        15
          RemoteNodeId                                        IfIndex
          72:80:47.000580ffe1000000f2000000.ff150e130002.00   1245184
          RemoteAtmAddress                            Vpi  Vci
          47.000580ffe1000000f2150e13.ff150e130002.00 0    33
```

```
switchaz2::configuration atmroute> display atmr pnni svccrcc
 Node SvccRccIx Version        HelloState    RcvHellos XmtHellos
 2    0         version1point0 twoWayInside  18        18
          RemoteNodeId                                        IfIndex
          72:80:47.000580ffe1000000f3000000.ff1a2e760002.00   1179648
          RemoteAtmAddress                            Vpi  Vci
          47.000580ffe1000000f31a0519.ff1a2e760002.00 0    33
```

NOTE

In *ForeThought* 6.0, policing is disabled by default on NNI links. When configuring PNNI, it is recommended that you leave the policing off and that you disable the policing on NNI links on any third-party switches that are part of the network. If you enable policing on the NNI links, the SVCC RCCs would be subject to policing and cells could be dropped on the SVCC RCCs.

3. Check that a horizontal link has been established between LGNs by using the following command on each LGN. If the horizontal link has been established properly, the `HelloState` shows `twoWayInside` and the `LinkType` shows `horizontalLinkToFromLgn`.

```
switchaz2::configuration atmroute> display atmr pnni link 2
 Node PortId     HelloState    LinkType                 Version        SvccRccIx
 2    0x30000000 twoWayInside  horizontalLinkToFromLgn  version1point0 0
      RmtNodeId                                           RmtPortId   DerAggrTk
      72:80:47.000580ffe1000000f2000000.ff150e130002.00   0x30000000  0x0
```

CHAPTER 7 Signalling

This chapter contains information that pertains to the signalling portion of *ForeThought* software. This signalling information is described in the following sections:

7.1 Virtual UNI

Virtual UNIs are supported across a physical interface by assigning each end user, one or more virtual path connections on its own interface and then using a VPCI group index identifying the VPCI-VPI-port mapping for each of the virtual UNI signalling interfaces on that physical interface.

One of the applications for virtual UNIs is in the DSL (Digital Subscriber Line) environment. DSL devices can have a dedicated signalling channel and dedicated virtual path to an ATM switch. As illustrated in Figure 7.1, DSL-to-ATM networks typically have a few DSL devices, aggregated by a DSLAM. A DSLAM (Digital Subscriber Line Access Multiplexer) is a device which takes all user VPCs and combines these onto one or more connections to an ATM switch by relaying user cells based only on their VPI. In other words, a DSLAM is a VP-Mux devoid of signalling interfaces.

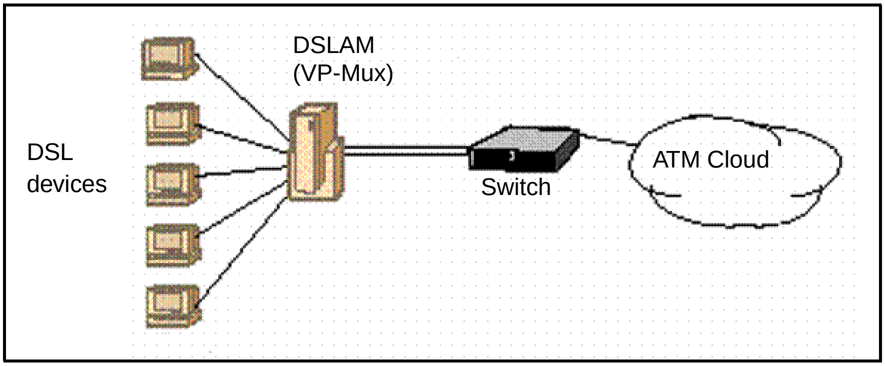

Figure 7.1 - DSL-to-ATM Network Connection

In most cases, virtual UNIs have VCs allocated on VP=0. This means that all end users (DSL devices) are signalling and requesting SVCs on VP=0. The VP-Mux maps each of the end users to a unique VP on the switch side. Virtual UNI support requires the switch to understand that setup messages received on a non-zero signalling path must be responded to as though all the devices were on path zero. The ATM switch allocates virtual circuits on the VPs on which it receives the Setup message. However, it translates the VPCI-VPI=0 when reporting it back to the end user in the Connect message.

In order for the switch to support VP-Muxes, it supports multiple virtual UNIs on the single physical interface to the VP-Mux. A unique VPI on the interface between the DSLAM and the switch is used to distinguish between individual end users. Switches send and receive signalling messages on a unique VPI associated with the individual user devices.

The switch maintains a translation table between VPCI and VPI. For the end user, the VPCI is equal to the VPI. The VPCI remains constant in the signalling message as it travels between the user and switch. The VPI associated with the VPCI is used while the switch allocates a virtual circuit upon receiving a Setup message. Similarly, the VPCI associated with the VPI on which the circuit is allocated is used while sending the Connect message back to the end user. The switch maintains two tables:

- VPCI mapping table (VMT): contains information for a single VPCI-VPI-port mapping where each entry in the VMT is assigned a unique map index.
- VPCI mapping group table: contains a listing of all VPCI map indexes where each entry in the VPCI group table is assigned a unique group index. This group index must be used when creating the signalling interfaces on the switch. This enables VPCI-VPI translation in the signalling messages.

The following procedure outlines the steps necessary to create virtual UNI connections:

1. Create one or more VPCI-to-VPI mappings using the **conf signalling vpcitable vpcimap new** <*vpcimap-index*>[**-vpci** <*vpci*>] [**-vpi** <*vpi*>] [**-port** <*port*>] AMI command.
2. Create a VPCI group for one or more of the VPCI-to-VPI mappings created in step 1. Use the **conf signalling vpcitable vpcigroup new** <*vpcigroup-index*> <*vpcimap-index*> AMI command.
3. Create the signalling channel using the **conf signalling new** <*port*> <*vpi*> AMI command and specify the assigned VPCI group index created in step 2.

For detailed information on using the above AMI commands, see Chapter 9 in Part 2 of the *AMI Configuration Commands Reference Manual*.

The following example shows how a single VPCI-VPI mapping can be achieved for three virtual UNIs using the VPCI mapping table (VMT). In Figure 7.2, a switch is connected to the VP-Mux and three end user devices are connected to the VP-Mux.

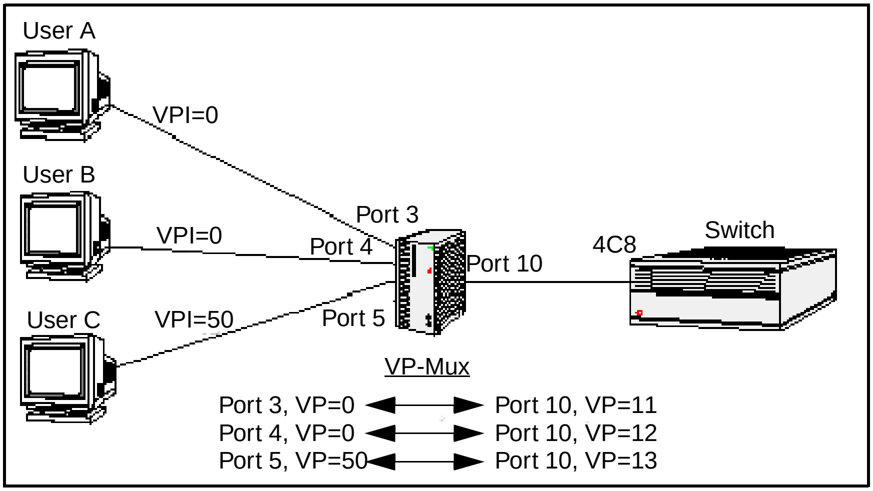
User A
VPI=0
User B
VPI=0
User C
VPI=50
Port 3
Port 4
Port 5
Port 10
4C8
Switch
VP-Mux
Port 3, VP=0 ⟷ Port 10, VP=11
Port 4, VP=0 ⟷ Port 10, VP=12
Port 5, VP=50 ⟷ Port 10, VP=13

Figure 7.2 - Virtual UNI Network Configuration

The connection setup procedure for the above example is as follows:

1. User A, attached to port 3 of the VP-Mux, sends a setup request out on VPI=0. A specific VCI is not required since the switch allocates any available VCI.
2. The VP-Mux performs ingress ports-to-VP mapping on egress port 10 and forwards the message to the switch on VPI=11, VCI=5.
3. The switch determines that VCI=53 is available within VPI=11 and assigns VPI=11, VCI=53 to the end user.
4. The switch maps VPI=11 to VPI=0 and sends a Connect message informing the end user to allocate a connection on VPI=0, VCI=53.
5. The VP-Mux performs VP-to-port mapping and forwards the message out port 3 over VPI=0, VCI=5 to the end user.
6. User A receives the Connect message and opens a circuit on VPI=0, VPI=53.

The steps for configuring the switch for the virtual UNI example illustrated in Figure 7.2 is as follows:

1. Create the VPCI-VPI-port mappings for the VPCI map table by entering the following AMI commands:

```
conf signalling vpcitable vpcimap new 101 -vpci 0 -vpi 11 -port 4C8

conf signalling vpcitable vpcimap new 102 -vpci 0 -vpi 12 -port 4C8

conf signalling vpcitable vpcimap new 103 -vpci 50 -vpi 13 -port 4C8
```

You can simplify the above configuration by entering the required parameter values and taking the default values for the optional parameters. For example:

```
conf signalling vpcitable vpcimap new 104

conf signalling vpcitable vpcimap new 105 -vpci 50
```

2. Create a VPCI group for the VPCI-VPI-port mappings by entering the following AMI commands:

```
conf signalling vpcitable vpcigroup new 1001 101

conf signalling vpcitable vpcigroup new 1002 102

conf signalling vpcitable vpcigroup new 1003 103
```

If you entered the simplified configuration, you would enter the following instead:

```
conf signalling vpcitable vpcigroup new 1004 104

conf signalling vpcitable vpcigroup new 1005 105
```

3. Create the signalling channels by entering the following AMI commands:

```
conf signalling new 4C8 11 -vpcigroupindex 1001

conf signalling new 4C8 12 -vpcigroupindex 1002

conf signalling new 4C8 -vpcigroupindex 1003
```

If you entered the simplified configuration, you would enter the following instead:

```
conf signalling new 4C8 11 -vpcigroupindex 1004

conf signalling new 4C8 12 -vpcigroupindex 1004

conf signalling new 4C8 -vpcigroupindex 1005
```

NOTE

For detailed information on the virtual UNI commands, see Chapter 9 in Part 2 of the *AMI Configuration Commands Reference Manual.*

The following must be taken into consideration when configuring virtual UNIs:

- There is a maximum configurable limit on the number of privates UNIs that can be created. The maximum limit depends on the switch platform: HA/CF processors allow a maximum of 192 signalling interfaces and P5/P6 processors allow a maximum of 600 signalling interfaces.
- A large number of NNI signalling interfaces is not addressed by virtual UNI support.
- Disable all debug tracing on the switch while using large number of private UNI signalling interfaces.
- The default control port VC space is 0 to 1024 for the *ASX*-200BX, *ASX*-1000, 0 to 8192 for the *ASX*-4000, and 0 to 4091 for the *ESX*-3000. Ensure that the VC space is sufficient to support the number of UNIs on the switch. If it is not sufficient, modify the maximum VCI number to be supported on the control port using the **conf vpt modify** <*port*> <*vpi*> **-maxvci** <*vci*> AMI command. See Chapter 20 in Part 2 of the *AMI Configuration Commands Reference Manual.*
- VC space on the switch: By default, 511 VCCs are allocated per VP and in turn to each signalling interface. For a large number of UNIs, it is recommended to configure VPs with a smaller VC range between 0-40.
- If end systems can be configured to run signalling and ILMI protocol (5 and 16, respectively) on non-reserved VCCs, these can be configured on, for example, VCC 32 and 33, respectively. In this case, the VC range per VP can be set between 32-40 with 6 VCCs used for other applications on the end systems.

7.2 VCI Allocation Range

The VCI allocation range is the range of VCI values in a path within which a signalling channel will allocate VCs. This range defaults to the VCI-range of the virtual path containing the signalling channel (also referred to as the *containing path*).

NOTE

If the signalling channel is a link-scope signalling channel, then this range applies to the containing path and all other paths within which the signalling channel has allocated connections.

A VCI allocation range can be specified when a signalling channel is created using the ATM Management Interface (AMI) command **`conf signalling new`**. The range is computed differently depending on whether or not you have ILMI running.

7.2.1 Determining the VCI Allocation Range with ILMI Down

If ILMI is not operational and if the VCI allocation range of the signalling channel is specified, then the range is determined as an intersection of the following:

- the VCI-range of the path containing the signalling channel
- the range specified for the signalling channel

For example, if you create a signalling channel with ILMI down on a VP that has no VCIs reserved, and you specify a VCI range:

```
conf signalling new 1A3 65 -ilmi down -minvci 70 -maxvci 200
```

the actual range is computed as an intersection based on the range you enter (70 - 200) and the range available in that VP (1 - 511).

To display what the actual VCI range is, enter the following command:

```
conf signalling show atmranges
```

```
         Admin  Admin   Oper   Oper  Admin      Oper     Oper
Port VPI MinVCI MaxVCI MinVCI MaxVCI MaxVPI MaxSvccVPI MaxSvpcVPI SvpCapable
1A1    0     32    511     32    511   1022        N/A        N/A        N/A
1A2    0     32    511     32    511   1022        N/A        N/A        N/A
1A3    0     32    511     32    511   1022        N/A        N/A        N/A
1A3   65     70    200     70    200   1022        N/A        N/A        N/A
1A4    0     32    511     32    511   1022        N/A        N/A        N/A
1CTL   0     32   1023     32   1023      0        N/A        N/A        N/A
```

In this display, the range that you have requested is shown in the `Admin MinVCI` and `Admin MaxVCI` fields. The range that is computed and allowed is displayed in the `Oper MinVCI` and `Oper MaxVCI` fields. In this example, no VCIs are reserved on VP 65, so the requested range of 70 - 200 is allowed.

In another example, if there are VCIs already reserved on VP 65, as shown below in the `MinVCI` and `MaxVCI` fields:

`configuration vpt show`

```
 Input      Output
 Port  VPI  Port  VPI   ResBW  CurBW MinVCI MaxVCI  VCs  Protocol
 1A1     0  terminate     N/A  37.3K      1    511   35  pvc
 1A2     0  terminate     N/A   0.8K      1    511    6  pvc
 1A3     0  terminate     N/A   0.8K      1    511    6  pvc
 1A3    65  terminate     N/A   0.8K      1    120    6  pvc
 1A4     0  terminate     N/A   0.8K      1    511    6  pvc
 1CTL    0  terminate     N/A  45.4K      1   1023   85  pvc
 originate  1A1     0     N/A   1.7K      1    511    7  pvc
 originate  1A2     0     N/A   0.8K      1    511    6  pvc
 originate  1A3     0     N/A   0.8K      1    511    6  pvc
 originate  1A3    65     N/A   0.8K      1    120    6  pvc
 originate  1A4     0     N/A   0.8K      1    511    6  pvc
Press return for more, q to quit: q
```

and you enter the following:

`conf signalling new 1A3 65 -ilmi down -minvci 70 -maxvci 200`

then the actual range is computed as an intersection based on the range you enter (70 - 200) and the range available in that VP (1 - 120).

To see what the actual VCI range is, enter the following command:

`conf signalling show atmranges`

```
         Admin  Admin   Oper   Oper  Admin      Oper     Oper
Port VPI MinVCI MaxVCI MinVCI MaxVCI MaxVPI MaxSvccVPI MaxSvpcVPI SvpCapable
1A1    0     32    511     32    511   1022        N/A        N/A        N/A
1A2    0     32    511     32    511   1022        N/A        N/A        N/A
1A3    0     32    511     32    511   1022        N/A        N/A        N/A
1A3   65     70    200     70    120   1022        N/A        N/A        N/A
1A4    0     32    511     32    511   1022        N/A        N/A        N/A
1CTL   0     32   1023     32   1023      0        N/A        N/A        N/A
```

In this example, since VCIs 1 - 120 are already reserved on VP 65, the range of 70 - 120 is the actual range allowed.

7.2.2 Determining the VCI Allocation Range with ILMI Up

If ILMI is operational, then the peer switch's VCI allocation range is obtained by reading its ILMI MIB variable `atmfAtmLayerMaxVccs`. The VCI allocation range of the signalling channel is then determined as an intersection of the following:

- the VCI-range of the path containing the signalling channel
- the range specified for the signalling channel (if applicable)
- the VCI-range of the peer signalling entity

For example, if your peer has no VCIs reserved, and you create a signalling channel with ILMI up on a VP that has no VCIs reserved, and you specify a VCI range:

```
conf signalling new 1A3 65 -ilmi up -minvci 70 -maxvci 200
```

the actual range is computed as an intersection based on the range you enter (70 - 200), the range available in that VP (1 - 511), and the range of the peer signalling entity (32 - 511).

To display what the actual VCI range is, enter the following command:

```
conf signalling show atmranges
```

```
         Admin  Admin   Oper   Oper  Admin      Oper      Oper
Port VPI MinVCI MaxVCI MinVCI MaxVCI MaxVPI MaxSvccVPI MaxSvpcVPI SvpCapable
1A1    0     32    511     32    511   1022        N/A        N/A        N/A
1A2    0     32    511     32    511   1022        N/A        N/A        N/A
1A3    0     32    511     32    511   1022        N/A        N/A        N/A
1A3   65     70    200     70    200   1022        N/A        N/A        N/A
1A4    0     32    511     32    511   1022        N/A        N/A        N/A
1CTL   0     32   1023     32   1023      0        N/A        N/A        N/A
```

In this display, the range that you have requested is shown in the `Admin MinVCI` and `Admin MaxVCI` fields. The range that is computed and allowed is displayed in the `Oper MinVCI` and `Oper MaxVCI` fields. In this example, no VCIs are reserved on VP 65 and the signalling peer supports the range of 32 - 511, so the requested range of 70 - 200 is allowed.

In another example, if the peer supports the range of 32 - 511, but there are VCIs already reserved on VP 65, as shown below in the `MinVCI` and `MaxVCI` fields:

```
configuration vpt show
```

```
 Input      Output
 Port  VPI  Port  VPI   ResBW  CurBW MinVCI MaxVCI  VCs  Protocol
 1A1      0  terminate     N/A  37.3K      1    511   35  pvc
 1A2      0  terminate     N/A   0.8K      1    511    6  pvc
 1A3      0  terminate     N/A   0.8K      1    511    6  pvc
 1A3     65  terminate     N/A   0.8K      1    120    6  pvc
 1A4      0  terminate     N/A   0.8K      1    511    6  pvc
 1CTL     0  terminate     N/A  45.4K      1   1023   85  pvc
 originate  1A1     0     N/A   1.7K      1    511    7  pvc
 originate  1A2     0     N/A   0.8K      1    511    6  pvc
 originate  1A3     0     N/A   0.8K      1    511    6  pvc
 originate  1A3    65     N/A   0.8K      1    120    6  pvc
 originate  1A4     0     N/A   0.8K      1    511    6  pvc
Press return for more, q to quit: q
```

and you enter the following:

```
conf signalling new 1A3 65 -ilmi up -minvci 70 -maxvci 200
```

then the actual range is computed as an intersection based on the range you enter (70 - 200), the range available in that VP (1 - 120), and the range supported by the peer (32 - 511).

To see what the actual VCI range is, enter the following command:

```
conf signalling show atmranges
```

```
         Admin  Admin   Oper   Oper  Admin      Oper     Oper
Port VPI MinVCI MaxVCI MinVCI MaxVCI MaxVPI MaxSvccVPI MaxSvpcVPI SvpCapable
1A1    0     32    511     32    511   1022        N/A        N/A        N/A
1A2    0     32    511     32    511   1022        N/A        N/A        N/A
1A3    0     32    511     32    511   1022        N/A        N/A        N/A
1A3   65     70    200     70    120   1022        N/A        N/A        N/A
1A4    0     32    511     32    511   1022        N/A        N/A        N/A
1CTL   0     32   1023     32   1023      0        N/A        N/A        N/A
```

In this example, since the peer supports VCIs 32 - 511, but VCIs 1 -120 are already reserved on VP 65, the range of 70 - 120 is the actual range allowed.

In a third example, if the peer only supports the range of 32 - 180, and VCIs 1 - 120 are already reserved on VP 65, and you enter the following:

```
conf signalling new 1A3 65 -ilmi up -minvci 70 -maxvci 200
```

then the actual range is computed as an intersection based on the range you enter (70 - 200), the range available in that VP (1 - 120), and the range supported by the peer (32 - 180).

To see what the actual VCI range is, enter the following command:

```
conf signalling show atmranges
```

```
         Admin  Admin   Oper   Oper  Admin      Oper       Oper
Port VPI MinVCI MaxVCI MinVCI MaxVCI MaxVPI MaxSvccVPI MaxSvpcVPI SvpCapable
1A1    0     32    511     32    511   1022        N/A        N/A        N/A
1A2    0     32    511     32    511   1022        N/A        N/A        N/A
1A3    0     32    511     32    511   1022        N/A        N/A        N/A
1A3   65     70    200     70    120   1022        N/A        N/A        N/A
1A4    0     32    511     32    511   1022        N/A        N/A        N/A
1CTL   0     32   1023     32   1023      0        N/A        N/A        N/A
```

In this example, since the peer supports VCIs 32 - 180, and VCIs 1 - 120 are already reserved on VP 65, the range of 70 - 120 is the actual range allowed.

7.3 Signalling Scope

A signalling channel can be configured to control either VCs within its own VP (VP-scope, the default) or to control VCs in its own VP, in addition to controlling VCs in other VPs on the link (link-scope).

The following criteria must be satisfied in order to use link-scope signalling channels:

- There can be only one link-scope signalling channel on a given link.
- A link-scope signalling channel must be created within VP 0 on the given link.
- The path containing the link-scope signalling channel must be an elastic path.
- A backplane signalling channel in an *ASX*-1000, *ASX*-1200, or a *TNX*-1100 cannot be a link-scope signalling channel.

7.3.1 VC-Space

The VC-space of a signalling channel is the VPI / VCI space within which a signalling channel can allocate connections. This is determined by the allocation scope of the signalling channel and the VCI allocation range (see Section 7.2) for the signalling channel.

The VC-space of a VP-scope signalling channel is the VCI allocation range of the signalling channel applied only to the path that contains the signalling channel.

The VC-space of a link-scope signalling channel is comprised of the aggregate VCI allocation range of the signalling channel applied to all paths controlled by the signalling channel (i.e., the path containing the signalling channel and all other paths in which the signalling channel has allocated connections).

7.3.2 Dynamic Paths

A link-scope signalling channel allocates new virtual channel connections based on the availability of virtual channel connections within the VC-space of the link-scope signalling channel at that time. A link-scope signalling channel allocates connections in a new originating/terminating path (called a *dynamic path*, because signalling uses such a path on the fly) when its current VC-space is exhausted.

NOTE

Every time a link-scope signalling channel uses a new path for allocating connections, its VC-space increases.

To avoid conflicts of ownership of the path, a link-scope signalling channel never uses an existing provisioned terminating/originating path (apart from the path that contains the signalling channel) to allocate virtual channel connections.

Dynamic paths inherit the characteristics of the virtual path that contains the signalling channel (called the *containing path*). Therefore, dynamic paths have the same VCI-range as the containing path and are elastic paths.

To see which paths are dynamic paths, look at the `Protocol` field associated with each path under the AMI command **`conf vpt show`**. This field displays `q2931` for dynamic paths and displays `pvc` for provisioned terminating/originating paths.

A dynamic path is controlled only by the link-scope signalling channel that uses it. Therefore, a dynamic path is not open to any provisioning such as the creation of a PVC connection, SPANS signalling path, etc.

7.4 Signalling Channel Auto Configuration Procedures

This section provides an overview of signalling channel auto configuration and describes some rules for configuring signalling interfaces. Auto configuration requires that ILMI is `up` on that signalling interface.

7.4.1 Overview of Signalling Channel Auto Configuration

Signalling support for UNI 3.0, UNI 3.1, and UNI 4.0 is provided in *ForeThought* 5.2.x and greater. The switch can be configured to detect which interfaces are UNI 3.0, which are UNI 3.1, and which are UNI 4.0 automatically. The switch also translates signalling messages between interfaces of different types.

When a signalling channel is created using **`conf signalling new -version auto`**, then the creation is delayed until a response is obtained from the peer or until the ILMI keep alive requests time out. Based on the peer response, the appropriate SSCOP stack is initialized. If the peer does not support the `atmfAtmLayerUniVersion` variable, then the switch software defaults to UNI 3.0. Similarly, if the ILMI keep alive requests time out, then the switch software defaults to UNI 3.0. The switch periodically checks the version supported on the peer by querying the `atmfAtmLayerUniVersion` variable. If the peer version has changed since the last query, the signalling stack is recreated to reflect the change.

The following two tables summarize the above information. Table 7.1 highlights the actions taken based on the configurations of the FORE switches on both sides of an interface. *ForeThought* versions prior to *ForeThought* 4.1.x always default to UNI 3.0. Both UNI 3.1 and UNI 4.0 use SSCOP version 31.

Table 7.1 - Action Taken Based on Both Switches' Signalling Channel Configurations

Configured Version	Peer's Configured Version	Action Taken
UNI 3.0	UNI 3.0	SSCOP 30 stack initialized
	UNI 3.1	SSCOP 30 stack initialized; error message displayed on syslog
	UNI 4.0	SSCOP 30 stack initialized; error message displayed on syslog
	PNNI 1.0	SSCOP 30 stack initialized; error message displayed on syslog
	Auto	SSCOP 30 stack initialized; peer changes version to UNI 3.0
UNI 3.1	UNI 3.0	SSCOP 31 stack initialized; error message displayed on syslog
	UNI 3.1	SSCOP 31 stack initialized
	UNI4.0	SSCOP 31 stack initialized; error message displayed on syslog
	PNNI 1.0	SSCOP 31 stack initialized; misconfiguration
	Auto	SSCOP 31 stack initialized; peer changes version to UNI 3.1
PNNI 1.0	UNI 3.0	SSCOP 31 stack initialized; error message displayed on syslog
	UNI 3.1	SSCOP 31 stack initialized; misconfiguration
	UNI4.0	SSCOP 31 stack initialized; misconfiguration
	PNNI 1.0	SSCOP 31 stack initialized
	Auto	SSCOP 31 stack initialized; peer changes version to PNNI 1.0 if it supports PNNI
UNI 4.0	UNI 3.0	SSCOP 31 stack initialized; error message displayed on syslog
	UNI 3.1	SSCOP 31 stack initialized; error message displayed on syslog
	UNI4.0	SSCOP 31 stack initialized
	PNNI 1.0	SSCOP 31 stack initialized; misconfiguration
	Auto	SSCOP 31 stack initialized; peer changes version to UNI 4.0
Auto	UNI 3.0	SSCOP 30 stack initialized
	UNI 3.1	SSCOP 31 stack initialized
	UNI4.0	SSCOP 31 stack initialized
	PNNI 1.0	SSCOP 31 stack initialized
	Auto	Both stacks initialized as SSCOP 31 if both sides support UNI 3.1 or UNI 4.0; otherwise, both sides are initialized to SSCOP 30. If both sides support PNNI then, both sides are initialized to SSCOP 31 and the version is set to PNNI 1.0.

An alternative way of looking at the auto configuration procedure is to look at the `atmfAtmLayerUniVersion` variable. Table 7.2 shows actions taken based on the `atmfAtmLayerUniVersion` variable supported by the peer switch.

Table 7.2 - Action Taken Based on the Peer's Supported MIB Variable

Configured UNI Version	atmfAtmLayer UNI Version	Action Taken
UNI 3.0	Version 3.0	SSCOP 30 stack initialized
	Version 3.1	SSCOP 30 stack initialized; error message displayed on syslog
	Version 4.0	SSCOP 31 stack initialized; error message displayed on syslog
	Version 2.0	SSCOP 30 stack initialized; version 2.0 is not supported and error message displayed on syslog
	MIB variable not supported	SSCOP 30 stack initialized
UNI 3.1	Version 3.0	SSCOP 31 stack initialized; error message displayed on syslog
	Version 3.1	SSCOP 31 stack initialized
	Version 4.0	SSCOP 31 stack initialized; error message displayed on syslog
	Version 2.0	SSCOP 30 stack initialized; version 2.0 is not supported and error message displayed on syslog
	MIB variable not supported	SSCOP 31 stack initialized; error message displayed on syslog
UNI 4.0	Version 3.0	SSCOP 30 stack initialized
	Version 3.1	SSCOP 30 stack initialized; error message displayed on syslog
	Version 4.0	SSCOP 31 stack initialized
	Version 2.0	SSCOP 30 stack initialized; version 2.0 is not supported and error message displayed on syslog
	MIB variable not supported	SSCOP 30 stack initialized
Auto	Version 3.0	SSCOP 30 stack initialized
	Version 3.1	SSCOP 31 stack initialized
	Version 4.0	SSCOP 31 stack initialized
	Version 2.0	SSCOP 30 stack initialized; version 2.0 is not supported and error message displayed on syslog
	MIB variable not supported	SSCOP 30 stack initialized

7.4.2 Rules for Signalling Channel Auto Configuration

In *ForeThought* 5.0.x and greater, there are two rules for signalling channel auto configuration. The first deals with specifying the signalling interface type and signalling interface version. The second rule concerns specifying the signalling scope and mode.

7.4.2.1 Specifying the Type and Interface Version

In AMI, the signalling interface type and signalling interface version are now treated as a pair. The pertinent portion of the AMI syntax is listed here for reference:

```
myswitch::configuration signalling> new <port> <vpi>
             [-version (auto | uni30 | uni31 | pnni10 | uni40)]
             [-type (auto | publicUNI | IISP | privateNNI | privateUNI)]
```

The type and version either both have to be `auto` or both have to be given a parameter other than `auto`. Specifying the type as `auto` and hard configuring the version or vice-versa is not allowed. If a type is not entered, then the type parameter defaults to `auto`. Similarly, the version parameter defaults to `auto` if a version is not entered.

The one exception to this rule (to allow backward compatibility) is for the type to be entered as `publicUNI` and the version to be entered as `auto`. Table 7.3 shows the valid combinations.

Table 7.3 - Valid Type and Version Combinations

Type	Version
auto	auto
privateUNI	uni30
privateUNI	uni31
privateUNI	uni40
publicUNI	auto[1]
publicUNI	uni30
publicUNI	uni31
publicUNI	uni40
IISP	uni30
IISP	uni31
IISP	uni40
privateNNI	pnni10

[1] This is the only exception to the rule.

7.4.2.1.1 Examples of Valid Configurations

The following are examples of valid configurations to enter into AMI:

```
        conf sig new 1a1 0
        conf sig new 1a1 0 -type privateUNI -version uni30
        conf sig new 1a1 0 -type privateUNI -version uni31
        conf sig new 1a1 0 -type privateUNI -version uni40
        conf sig new 1a1 0 -type publicUNI
        conf sig new 1a1 0 -type publicUNI -version auto
        conf sig new 1a1 0 -type publicUNI -version uni30
        conf sig new 1a1 0 -type publicUNI -version uni31
        conf sig new 1a1 0 -type publicUNI -version uni40
        conf sig new 1a1 0 -type IISP -version uni30
        conf sig new 1a1 0 -type IISP -version uni31
        conf sig new 1a1 0 -type IISP -version uni40
        conf sig new 1a1 0 -type privateNNI -version pnni10
```

The first example is the default that is most often used when configuring a signalling interface. Using this specification allows the interface to auto configure to any possible valid combination like `privUNI/uni30`, `privUNI/uni31`, `privUNI/uni40`, `ftpnni/uni30`, `ftpnni/uni31`, `ftpnni/uni40`, or `privateNNI/pnni10`. This is the most preferred input.

The fifth and sixth examples illustrate the exception to the rule.

7.4.2.1.2 Examples of Invalid Configurations

Table 7.4 shows examples of invalid configurations to enter into AMI and the reasons why they are invalid.

Table 7.4 - Invalid Type and Version Combinations

Invalid Combination	Reason
conf sig new 1a1 0 -version uni30	The type is not entered (auto).
conf sig new 1a1 0 -type auto -version uni30	The type is auto.
conf sig new 1a1 0 -version uni31	The type is not entered (auto).
conf sig new 1a1 0 -type auto -version uni31	The type is auto.
conf sig new 1a1 0 -version uni40	The type is not entered (auto).
conf sig new 1a1 0 -type auto -version uni40	The type is auto.
conf sig new 1a1 0 -version pnni10	The type is not entered (auto).
conf sig new 1a1 0 -type auto -version pnni10	The type is auto.
conf sig new 1a1 0 -type privateUNI	The version is not entered (auto).
conf sig new 1a1 0 -type privateUNI -version auto	The version is auto.
conf sig new 1a1 0 -type privateUNI -version pnni10	pnni10 is an NNI protocol.
conf sig new 1a1 0 -type publicUNI -version pnni10	pnni10 is an NNI protocol.
conf sig new 1a1 0 -type IISP	The version is not entered (auto).
conf sig new 1a1 0 -type IISP -version auto	The version is auto.
conf sig new 1a1 0 -type IISP -version pnni10	IISP and pnni10 are conflicting NNI protocols.
conf sig new 1a1 0 -type privateNNI	The version is not entered (auto).
conf sig new 1a1 0 -type privateNNI -version auto	The version is auto.
conf sig new 1a1 0 -type privateNNI -version uni30	The NNI type and the UNI version are conflicting protocols.
conf sig new 1a1 0 -type privateNNI -version uni31	The NNI type and the UNI version are conflicting protocols.
conf sig new 1a1 0 -type privateNNI -version uni40	The NNI type and the UNI version are conflicting protocols.

7.4.2.2 Specifying the Scope and Mode

In AMI, the signalling scope (designated in AMI as **-sig_alloc**) and the signalling mode (designated in AMI as **-sig_mode**) are now treated as a pair. The pertinent portion of the AMI syntax is listed here for reference:

```
myswitch::configuration signalling> new <port> <vpi>
  protocol options:
         [-sig_alloc (vp | link | auto)]
         [-sig_mode (nonAssoc | vpAssoc | auto)]
```

The scope and mode either both have to be `auto` or both have to be given a parameter other than `auto`. Specifying the scope as `auto` and hard configuring the mode or vice-versa is not allowed. If a scope is not entered, then the scope parameter defaults to `auto`. Similarly, the mode parameter defaults to `auto` if a mode is not entered.

Since scope and mode are not negotiated based on the peer signalling entity's status, if the scope and mode are hard configured on one side of a link, they should also be hard configured on the remote side.

Table 7.5 shows the valid combinations.

Table 7.5 - Valid Scope and Mode Combinations

Scope	Mode
autoScope	autoMode
vpScope	vpAssoc
vpScope	nonAssoc[1]
linkScope	nonAssoc

[1] This combination is valid as long as the **-type** is not configured as `auto` or `privateNNI`. If the **-type** is `auto`, the link could become a PNNI link, which would conflict with a **-mode** of `nonAssoc` and a **-scope** of `vpScope`.

7.4.2.2.1 Examples of Valid Configurations

The following are examples of valid configurations to enter into AMI:

```
conf sig new 1a1 0
conf sig new 1a1 0 -sig_alloc auto -sig_mode auto
conf sig new 1a1 0 -sig_alloc vp -sig_mode vpAssoc
conf sig new 1a1 0 -sig_alloc link -sig_mode nonAssoc
```

The first example is the default that is most often used when configuring a signalling interface. This is the most preferred mode since the auto configuration of the link and scope procedures can correctly determine what is valid for a given interface.

7.4.2.2.2 Examples of Invalid Configurations

Table 7.6 shows examples of invalid configurations to enter into AMI and the reasons why they are invalid.

Table 7.6 - Invalid Scope and Mode Combinations

Invalid Combination	Reason
conf sig new 1a1 0 -sig_mode vpAssoc	The scope is not entered (auto).
conf sig new 1a1 0 -sig_alloc auto -sig_mode vpAssoc	The scope is auto.
conf sig new 1a1 0 -sig_mode nonAssoc	The scope is not entered (auto).
conf sig new 1a1 0 -sig_alloc auto -sig_mode nonAssoc	The scope is auto.
conf sig new 1a1 0 -sig_alloc vp	The mode is not entered (auto).
conf sig new 1a1 0 -sig_alloc vp -sig_mode auto	The mode is auto.
conf sig new 1a1 0 -sig_alloc link	The mode is not entered (auto).
conf sig new 1a1 0 -sig_alloc link -sig_mode auto	The mode is auto.
conf sig new 1a1 0 -sig_alloc link -sig_mode vpAssoc	This is an invalid combination for an interface.
conf sig new 1a1 0 -sig_alloc vp -sig_mode nonAssoc	This is an invalid combination if the type is auto (default).
conf sig new 1a1 0 -type privateNNI -version pnni10 -sig_alloc vp -sig_mode nonAssoc	VP scope and nonAssoc mode are incompatible with the type privateNNI.

There is a potential invalid configuration that can occur because the auto configuration of the operating scope and mode depend on whether a path is an elastic path or not. A non-elastic path has bandwidth reserved for it from the link bandwidth at the time that the path is created (when the **-reserved** option in **conf vpt new** is used.) An elastic path does not have reserved bandwidth at creation time. Instead, it uses the residual bandwidth available on the link.

The invalid configuration can occur when a signalling interface on VPI 0 that has both the scope and mode configured as `auto` becomes a PNNI interface (either by the user explicitly specifying **-type privateNNI** or by specifying **-type auto**). In this case, if the path is an elastic path, then the operating scope becomes `link` and the operating mode becomes `nonAssoc`. If the path is a non-elastic path, then the operating scope becomes `vp` and the operating mode becomes `vpAssoc`.

Therefore, if a signalling interface exists between two switches on VPI 0, with one half of this path being elastic on one switch and one half being non-elastic on the other switch, then the interface comes up with incompatible modes on either side if it becomes a PNNI interface. This results in no calls being set up on this interface.

To prevent this invalid configuration, be sure that both sides of your PNNI interface are either elastic or non-elastic. If this invalid configuration occurs, simply delete one side of the path and recreate it to match the other side.

7.4.2.3 Specifying the User/Network Side

While *ForeThought* automatically detects the side depending on the type, version, and routing domain, there is one exception in the case of publicUNI. Unless the side is hard configured as `network,` a publicUNI will always come up as `user`. To avoid this, configure the network side first and then the user side will be automatically detected.

7.5 Allowable Combination of Traffic Parameters

7.5.1 PNNI 1.0/UNI 4.0

The following information applies to both signalled connections and to PNNI SPVCs.

7.5.1.1 Service Categories

The ATM service categories assigned to connections in PNNI 1.0 and UNI 4.0 are defined in the ATM Traffic Management Specification, Version 4.0 (TM 4.0). For PNNI 1.0 interfaces, *ForeThought* 5.2.x and greater supports the service categories defined in TM 4.0 with a few exceptions.

An explicit way of requesting a particular ATM service category is not provided in PNNI 1.0 and UNI 4.0 signalling. Instead, it must be derived from three pieces of information in the SETUP message.

- the broadband bearer class in octet 5 of the broadband bearer capability (BBC) information element
- the absence or presence of the ATM transfer capability (ATC) octet (octet 5a) in the BBC information element
- the value of the ATC, if present, and the absence or presence of the best effort indicator in octet 18 of the ATM Traffic Descriptor information element

Derivation of a service category from the above information is specified in Section A9.2 (Determination of ATM service Category) of the ATM User-Network Interface (UNI) Signalling Specification, Version 4.0 (UNI 4.0). However, *ForeThought* 5.2.x and greater does not support the following from Table A9-1 of this specification:

- The Transparent VP-Service in the BBC information element is not supported.

7.5.1.2 Allowable Combination of Traffic Parameters

The parameters specified in the BBC information element, the ATM traffic descriptor information element, the Extended QoS parameter information element, the End-to-End transit delay information element, and the QoS parameter information element of the SETUP message should be consistent.

Table A9-2 of UNI 4.0 shows the allowable combinations. The following exceptions apply to *ForeThought* 5.2.x and greater:

- In the Extended QoS parameters information element, for forward and backward Cell Loss Ratio (CLR), *ForeThought* 5.2.x and greater does not distinguish between the (CLP=0) and CLP (0+1) traffic streams.
- The PNNI 1.0 and UNI 4.0 specifications allow incomplete traffic contracts for VBR in four instances (two for real time VBR and two for non-real time VBR). Check note 7 and 8 of Table A9-2 of the UNI 4.0 specification. *ForeThought* 5.2.x and greater does not support these four combinations.

7.5.2 UNI 3.X

FORE Systems supports a subset of traffic contracts which are specified in Table F.1 of the UNI 3.1 specification. These traffic contracts are enforced on both signalled connections as well on PNNI SPVCs. The following are the allowable contracts supported by FORE's switch fabrics.

- CBR: PCR (0), PCR (0+1)
- CBR: PCR (0+1)
- VBR: PCR (0+1), SCR, MBS (0)
- VBR: PCR (0+1),SCR, MBS (0+1)
- UBR: PCR (0+1)

NOTE

Traffic contracts that do not completely characterize the traffic characteristics of a given service category are incomplete and not supported. (For example, a VBR contract that is received with an ATM traffic descriptor information element that does not specify SCR and MBS cell rates does not fully characterize a VBR contract. Such a contract is, therefore, rejected by the switch fabric).

Table 7.7 shows the subset of the UNI 3.1 allowable combination of traffic parameters supported by *ForeThought* 5.2.x and greater:

Table 7.7 - UNI 3.1 Allowable Combination of Traffic Parameters in *ForeThought* 5.2.x and Greater

Broadband Bearer Class	A	X	C	X	C	X	A	X	C	X
Traffic Type		CBR		&		&		CBR		&
Timing Required		Y		&&		&&		Y		&&
PCR (CLP=0)	S	S								
PCR (CLP=0+1)	S	S	S	S	S	S	S	S	S	S
SCR and MBS (CLP=0)			S	S						
SCR and MBS (CLP=0+1)					S	S				
Best Effort									S	S
Tagging	Y/N	Y/N	Y/N	Y/N	N	N	N	N	N	N
QoS Class	*	*	*	*	*	*	*	*	0	0

The following are used in the table above:

Y	Yes
N	No
S	Specified
Y/N	Either Yes or No is allowed.
*****	All QoS classes are supported.
&	The parameter is coded to either “no indication,” or “VBR,” or, for UNI 3.1, Octet 5a (Traffic Type / Timing Required) is absent; these 3 codings are treated as equivalent.
&&	The parameter is coded to either “no indication,” or “No,” or, for UNI 3.1, Octet 5a (Traffic Type / Timing Required) is absent; these 3 codings are treated as equivalent.

A blank entry indicates that the parameter is not present.

CHAPTER 8 Security

ForeThought software (that is version 5.0.x or greater) provides various forms of switch security. Security can be implemented by creating userids with different levels of privileges. The user login authentication can be implemented using Kerberos and SecurID. There are also security methods that prevent access to the switch, which include IP filtering and NSAP filtering. Each of these security methods is described in the following sections.

8.1 Kerberos Authentication

ForeThought 6.0.x and greater supports Kerberos V5 authentication. Kerberos is a secret-key network authentication protocol designed to authenticate requests for network resources. Kerberos was created and developed at Massachusetts Institute of Technology (MIT) to provide a means of verifying the identities of a workstation user or a network server, over an unsecure network.

Kerberos performs authentication as a trusted third party service (Kerberos server) using the Data Encryption Standard (DES) for encryption. This data encryption method protects communications between the server and the switch. When a user sends a request to the Kerberos server for access to a switch, the server issues a ticket to the user that links that user to the switch. This “ticket” is a record that contains the user’s identification and a temporary encryption key (session key). The session key is necessary to successfully use that ticket in an authentication exchange. The user receives this ticket and transmits it back to the server with the current time stamped. The session key, now shared by the user and server, is used to authenticate the user to the switch. The ticket and the ticket’s session key is known as a credential.

Kerberos prevents a user from logging into a switch via a DES encrypted telnet session with Kerberos authenticated session key exchange.

NOTE

Because Kerberos does not protect SNMP, ILMI, remote AMI, or *Element Manager* access to the switch, it is recommended that users either employ IP filtering as a selective mechanism to allow SNMP changes, or disable SETs from the network entirely.

Currently, the Kerberos AMI configuration commands are disabled for international versions of *ForeThought* software since the encryption technology used for Kerberos is not licensed for foreign distribution. The following message displays if a Kerberos command is executed on an international version of *ForeThought*:

```
myswitch::configuration security kerberos> show
Kerberos is not supported in international versions of ForeThought.
```

8.1.1 Kerberos Server

The Kerberos server can run on a UNIX system or on a Windows NT system. Each FORE switch has a defined set of authorized users. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information.) If a user requests a DES encrypted telnet session to a switch, that user is not permitted access to that switch until his or her identity is validated.

A Kerberos server maintains a database of user, server, and password information. When a user attempts to access a switch via a DES encrypted telnet session, the switch configured for Kerberos authentication verifies the session key and encrypts and decrypts messages sent between the switch and the telnet session.

8.1.2 Kerberos AMI Commands

The **`conf security kerberos`** AMI command allows Kerberos configuration on the switch. The Kerberos commands are described in detail in Part 2 of the *AMI Configuration Commands Reference Manual*.

8.1.3 Configuring Kerberos Authentication on a Switch

The following tasks should be completed to configure Kerberos authentication:

- Install the Kerberos server software
- Define the Kerberos realm name
- Configure the Kerberos server database
- Transfer the secret key file
- Set the time on the switch

NOTE

For detailed information on configuring Kerberos authentication on your switch, see Chapter 8 of Part 2 of the *AMI Configuration Commands Reference Manual*.

8.1.3.1 Installing the Kerberos Server Software

A Kerberos database server must be set up to implement the Kerberos authentication method for users of the switch. A Kerberos server and database program running on a network host is often referred to as the KDC (Key Distribution Center). Refer to the documentation that came with your Kerberos software for instructions about installing the software on the server.

8.1.3.2 Defining the Kerberos Realm

The Kerberos realm is an administrative domain containing a database of users and switches for that particular domain. Each Kerberos realm will have at least one Kerberos server.

8.1.3.3 Configuring the Kerberos Server Database

All users and switches in the defined Kerberos realm requiring Kerberos authentication must be registered in the Kerberos server database. This allows hosts and the Kerberos server in your realm to communicate and authenticate to each other. Kerberos uses the IP address and qualified host name to build a principal name. A principal is a string consisting of the name or type of service being offered, the name of the host offering this service, and the Kerberos realm name.

A key table (keytab) database (v5srvtab file) is generated for the principals entered into the Kerberos server. This file contains service keys known only by the KDC and the switch.

8.1.3.4 Transferring the Secret Key File

Once the `v5srvtab` file has been generated with all the secret keys for the switch and its interfaces, the file should be installed on the switch using FTP. Use the **`conf security kerberos v5srvtab install`** command to transfer the specified file from the server to the switch.

NOTE

This **`install`** command can be executed only by the users with `admin` privileges.

This key file is read only. Information is stored in the FLASH so that Kerberos configuration persists across reboots.

NOTE

Even though this information is stored in the FLASH, it cannot be accessed using any of the **`oper flash`** commands. The **`conf security kerberos show`** command must be used.

Authentication fails if the key file is not found, if the wrong file was installed, or if the file is corrupt. In these cases, an appropriate message is logged to the console, and access is denied.

8.1.3.5 Setting the Correct Time

For Kerberos authentication to work, it is important to set the correct time on the switch using the **oper date** AMI command. Kerberos does not utilize the time zone, but uses the switch's clock directory which is in GMT. The **oper date** command uses the time zone to convert the local time zone to GMT. The time set on the switch must be within five minutes of the time on the KDC (or to whatever time change defined on the KDC). Otherwise, authentication will fail since Kerberos will assume that someone is trying a replay attack. See Section 3.2 of the *ATM Management Interface (AMI) Manual* for more information.

8.1.3.6 Adding IP Address-to-Hostname Mappings

Once Kerberos is running, IP address-to-hostname mappings (for the local machine's interface) can be added or deleted from the CDB by entering the **configuration security kerberos host new** or **configuration security kerberos host delete** AMI commands. See Section 8.2.3 in Chapter 8 of Part 2 of the *AMI Configuration Commands Reference Manual*.

8.1.3.7 Establish an Encrypted Telnet Session

A request for a DES encrypted telnet session allows users to be authenticated by their Kerberos credentials before a Telnet session with a switch is established. Once the Kerberos server is set up and the secret key file has been installed on the switch, all users who are configured for Kerberos authentication will see a message when attempting to telnet into that switch.

8.1.4 An Example Login Using Kerberos Authentication

Once the Kerberos server is set up and the key file has been installed on the switch, users who are set up for Kerberos authentication will see a message when attempting to telnet into that switch. In the following example, a userid `eng` has been created on a switch that has the `v5srvtab` file in FLASH. The following is a transcript of how the user logs in, with the switch output in plain courier font and the user input in bold courier font:

```
telnet -x fishtank
Trying 172.19.4.20...
Connected to fishtank.
Escape character is '^]'.
[ Kerberos V5 accepts you as ``jsmith@FORE.COM'' ]

S_ForeThought_6.0.0 Development (1.29680+30d) (asx1000) (fishtank)
```

If Kerberos authentication fails, the following message displays:

```
Trying 172.19.4.20...
Connected to fishtank.
Escape character is '^]'.
Authentication negotiation has failed, which is required for encryption. Goodbye.
```

After Kerberos authentication, you are still prompted for your AMI userid. See Section 8.2 for more information on userids.

8.2 Configuring Userids

The network administrator creates and assigns a *userid* for each user or for a group of users via the **`conf security login new`** command in the ATM Management Interface (AMI). (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about this command.) A userid consists of the following:

- a method of login authentication
- a level of AMI command privileges
- a level of AMI access levels
- an optional password

On a new switch running *ForeThought* 5.0.x or greater, there are two separate default userids: `ami` and `asx`. Both are configured with the `local` authentication method, with `admin` privileges (meaning you are allowed to use all AMI commands), and `all` access (meaning you are allowed to login to the switch using all the possible methods). Both userids are assigned a null password.

The network administrator should configure userids for all people who will have access to each switch. Changing a userid only changes it in the switch's local `login` file. If the administrator wants to use the same set of userids on each switch in the network, he or she should configure the entire set of userids on any one switch. Then, he or she can backup the `login` file to a host using the **`conf security login backup`** command. Then, the administrator can copy the `login` file to each of the other switches in the network using the **`conf security login restore`** command on each switch. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about these commands.)

If the administrator wants to change, add, or delete one or more userids and propagate these changes to the other switches, he or she should use the same method. First, make all of the modifications on one switch, back up the `login` file to a host, and restore the `login` file to each of the other switches.

There are several different scenarios that can occur, such as the SecurID server being down or a userid not being listed on a particular switch. Table 8.1 shows the different login scenarios and the action taken by the switch for each. In this table, `port` refers to the access method assigned to the userid (e.g., telnet or serial port).

Table 8.1 - Possible Login Scenarios

Login Scenarios	**Action**
Userid not listed and SecurID server not accessible	Reject
Userid not listed; SecurID server accessible; and SecurID server rejects userid	Reject
Userid not listed; SecurID server accessible; and SecurID server accepts userid	Accept
Userid listed and port not OK	Reject
Userid listed; port OK; local authentication; and supplied password does not match	Reject
Userid listed; port OK; local authentication; and supplied password matches	Accept
Userid listed; port OK; SecurID authentication; SecurID server accessible; and SecurID server rejects userid	Reject
Userid listed; port OK; SecurID authentication; SecurID server accessible; and SecurID server accepts userid	Accept
Userid listed; port OK; SecurID authentication; SecurID server not accessible; and no local password	Reject
Userid listed; port OK; SecurID authentication; SecurID server not accessible; local password; and supplied password does not match	Reject
Userid listed; port OK; SecurID authentication; SecurID server not accessible; local password; and supplied password matches	Accept

8.2.1 Login Authentication Method

The network administrator can configure two different forms of login authentication: local authentication and SecurID authentication. The administrator may employ these methods for all users, or choose to employ the different methods for different users. Each method is described in the following sections.

8.2.1.1 Local Authentication

When a user is configured for local password authentication, he or she is prompted for a login ID (userid) and a password which is stored locally in the switch whenever he or she attempts to open an AMI session either via telnet or via the serial port. (The login password is not required if the user tries to open an AMI session via a remote switch or via *ForeView*.) After a validation check is made based on the scenarios listed in Table 8.1, an AMI session is started (provided that a local AMI session is not already running).

For specific information and examples of how to log in to the switch via telnet, the serial port, or a remote switch, see the *ATM Management Interface (AMI) Manual*. For information about logging in via *ForeView*, see the *ForeView Network Management User's Manual*.

8.2.1.2 SecurID Authentication

When a user is configured for SecurID authentication and he or she attempts to log in, the user is prompted for a login ID (userid) and a SecurID passcode. The two-part passcode consists of: a secret, memorized personal identification number (PIN) and the current code generated by the user's assigned SecurID token. (The passcode is not required if the user tries to open an AMI session via a remote switch or via *ForeView*.) After a validation check is made based on the scenarios listed in Table 8.1, an AMI session is started (provided that a local AMI session is not already running).

8.2.1.2.1 SecurID Protection on Switches

Security Dynamics ACE/server and client software[1] prevents a user from logging into AMI locally on a switch until the passcode entered by the user has been validated using an external security server. The server uses SecurID tokens to validate the identity of users, and allows access only to authorized users on valid clients (switches).

NOTE

Because SecurID does not protect SNMP, ILMI, remote AMI, or *ForeView* access to the switch, it is recommended that users either employ IP filtering as a selective mechanism to allow SNMP changes, or disable SETs from the network entirely.

[1] The client software is already provided on FORE ATM switches, but the server software must be purchased separately from Security Dynamics.

8.2.1.2.2 SecurID Passcode

This authentication method provides a high level of security because the SecurID passcode that allows access to the protected switches is comprised of two parts:

- a secret, memorized personal identification number (PIN)
- the current code generated by the user's assigned SecurID token

8.2.1.2.2.1 PIN Number

The PIN is known only by the user. It can be either alpha-numeric or strictly numerical, and can be either a fixed or variable length from 4 - 8 characters, depending on how the system administrator configures the server.

8.2.1.2.2.2 SecurID Tokens

The second part is a unique code from the SecurID token that only the user possesses and which cannot be counterfeited. Each authorized user on a protected system is assigned a SecurID token to use when accessing a protected switch. SecurID tokens are small, hand-held devices that use microprocessors to calculate and display random codes. These codes change at a specified interval, which is usually once every minute. The random code displayed on a user's token is the same code the server software has generated for that moment.

8.2.1.2.3 SecurID Server

The server can run on a UNIX system or on a Windows NT system. Each FORE switch has a defined set of authorized users. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information.) When a user is designated as someone who has SecurID authorization, that user is not permitted access to that switch until his or her identity is validated based on the scenarios listed in Table 8.1.

When a user attempts to login with a SecurID passcode, the SecurID software running on the switch verifies the passcode with the server; verifies the authenticity of the server so that no other machine can pretend to be the server in order to capture security data; and encrypts and decrypts messages sent between the switch and the server.

8.2.1.2.3.1 Slave Server

A backup, or slave server can be installed to ensure that authentication services are not interrupted, even if the server goes down. When the slave detects that the master has failed, the slave takes over authentication services. This failover is transparent to the user.

8.2.1.2.3.2 Server Database

The server's database includes records for all tokens, a list of switches to be protected, an audit trail of SecurID and administrative activity, and a list of users who are authorized to access each switch.

8.2.1.2.3.3 Data Encryption between the Server and Switches

Messages sent between the server and the switches are encrypted either using the DES algorithm or Security Dynamics proprietary encryption algorithm. The server can use either method, but all switches must use the same algorithm as the one configured on the server.

This data encryption method protects communications between the server and the switches because the first time the switch contacts the server, it receives a `node secret` file, which is a string of about 16 bytes. This string, which is known only to the server and this switch, is used in encrypting messages between the server and the switch. Additionally, communications between the master and slave servers and stored token information are encrypted. No one, including system administrators, can breach security by inspecting secured token data.

8.2.1.2.4 SecurID AMI Commands

The **`conf security login securid`** AMI command allows SecureID configuration on the switch. The SecureID commands are described in detail in Part 2 of the *AMI Configuration Commands Reference Manual*.

8.2.1.2.5 Configuring SecurID on a Switch

The following sections describe how to install SecurID on a switch.

8.2.1.2.5.1 Installing the Server Software

A SecurID server must be run to implement the authentication method for users of the switch. Refer to the Security Dynamics ACE/Server or Client user's manuals for instructions about installing the software on the server.

8.2.1.2.5.2 Transferring the Configuration File

As described earlier, the server and the switches need to maintain some common configuration parameters. The desired configuration information is specified in the `sdconf.rec` file when the server is installed. Once you have installed the server software, copy this file to the switch using the AMI command **`conf security login securid get`**. This command uses the FTP protocol to transfer the specified file from the specified server to the switch.

NOTE

This **`get`** command can be executed only by the users with `admin` privileges.

This configuration file is read and information is stored in the FLASH so that they persist across reboots. Additionally, the first time the switch contacts the server, it receives a `node secret` file, which is a string of about 16 bytes. This string, which is known only to the server and this switch, is used in encrypting messages between the server and the switch and is also stored in the FLASH.

NOTE

Even though this information is stored in the FLASH, it cannot be accessed using any of the **oper flash** commands. The **conf security login securid get**, **delete**, and **show** commands must be used.

If the configuration file is not found, if the wrong file was copied, or if the file is corrupted, the SecurID service does not work, an appropriate error message is logged to the console, and the user trying to log in is denied access.

8.2.1.2.5.3 Editing the Server Configuration File

Once SecurID is running, it possible to make changes to the `sdconf.rec` file for the server(s)/ clients. The file can be modified on the server itself by running **sdsetup** again and changing the desired parameters.

NOTE

After the changes are made on the server, use the **conf security login securid get** AMI command to copy the updated `sdconf.rec` file back to all of the switches that are using that file.

A sample of the contents of a `sdconf.rec` file and a description of the parameters follows:

```
Server License and Configuration

LICENSE CREATION: Mon July 27 15:06:08 1998
LICENSE ID: 96012648
ACE/Server VERSION: v 2.1.104
FILE OWNERSHIP: root
SETUID BIT: off
CLIENT RETRY: 5 times
CLIENT TIMEOUT: 5 sec
DES ENCRYPTION: allowed and enabled
DURESS MODE: not allowed
NetSP CLIENTS: not allowed
EXTENDED TACACS: disabled
TACACS PLUS: disabled
MASTER SERVER: linus
MASTER SERVER ADDRESS: 204.95.89.107
SLAVE SERVER: allowed but not configured
AUTHENTICATION SERVICE: securid
PORT NUMBER: 1024
```

```
ADDRESSES: By name in host file or name service
BAD PASSCODES: 3
RESPONSE DELAY: 2
TOKENS IN LICENSE: 25
LICENSE CONFIGURATION
This license was created for:
Fore Systems, Inc.
1000 FORE Drive
Warrendale, PA 15086
```

These parameters are defined as follows:

Parameter	Description
LICENSE CREATION	The date the ACE/server license was created.
LICENSE ID	The ACE/server license ID.
ACE/Server VERSION	The version number of the server software.
FILE OWNERSHIP	The account that owns the ACE/server files; i.e., that has permissions to run the ACE/server administration programs. **`root`** is the default, but the login of any other administrator can be specified here.
SETUID BIT	If set, allows anyone who can run the programs to run them with the permissions of the files' owner.
CLIENT RETRY	Specifies how many times a client attempts to establish communications with the server before reporting an error.
CLIENT TIMEOUT	Specifies how many seconds should elapse between attempts to establish client-server communications.
DES ENCRYPTION	Shows if DES or Security Dynamics encryption is used for client-server communications.
DURESS MODE	Allows a user to enter a special PIN to signal that he or she is being forced to log in by an unauthorized person.
NetSP CLIENTS	Used for IBM's Network Security Program.
EXTENDED TACACS	Indicates if the server supports XTACACS clients.
TACACS PLUS	Indicates if the server supports TACACS+ clients.
MASTER SERVER	The name of the master server.
MASTER SERVER ADDRESS	The IP address of the master server.
SLAVE SERVER	Indicates that the slave server may be installed. If it is installed, its name and IP address become part of the configuration record.
AUTHENTICATION SERVICE	The name of this authentication service as it appears in the /etc/services or in a NIS Services file.
PORT NUMBER	The UDP port number that has been assigned for the use of this service.

Parameter	Description
ADDRESSES	Indicates how network devices shall have the IP addresses resolved.
BAD PASSCODES	The number of failed login attempts with incorrect passcodes, after which the server puts the token associated with that userid into `Next Token Mode`. When the token is in this mode, the next time the same user tries to log in and gets the passcode correct, he or she is also prompted to enter the next token code; i.e., the one that appears next on the token after 60 seconds.
RESPONSE DELAY	Indicates how long an authentication request should be held (by the server) before a response is returned to the client.
TOKENS IN LICENSE	The number of tokens for this license.
LICENSE CONFIGURATION	Information about the owner of the license.

8.2.1.2.5.4 An Example Login Using SecurID

Once the SecurID server is set up and the configuration file has been copied to the switch, all users who are configured for securID will see a passcode prompt after the AMI login prompt when they attempt to log in to that switch. In the following example, a userid `eng` has been created on a switch that has the `sdconf.rec` file in FLASH and that has the MCA method for userid `eng` set to `securid`. The following is a transcript of how the user logs in, with the switch output in plain courier font and the user input in bold courier font:

```
localhost::> telnet fishtank
Trying 169.144.48.45 ...
Connected to bx02.
Escape character is ‘^]’.
S_ForeThought_6.0.0 (1.28144) (asx200bx) (fishtank)
login: eng <ENTER>
Enter PASSCODE: <PIN><Code on SecurID Token> <ENTER>
```

If the passcode entered by the user is accepted, an AMI session to the switch is started. If the server denies access to the user, the following message appears on the screen: `Login incorrect`. After three unsuccessful attempts, the telnet connection is torn down by the switch.

8.2.2 AMI Command Privileges

The network administrator can configure two different levels of AMI command privileges for userids: `admin` privileges and `user` privileges. Each method is described in the following sections.

8.2.2.1 Admin Privileges

A person whose userid is configured with `admin` privileges is allowed to access and use all of the AMI commands. This level of privileges should be reserved for the system administrator.

8.2.2.2 User Privileges

A person whose userid is configured with `user` privileges is allowed to access and use all AMI commands, except the following:

- **conf security login backup**
- **conf security login delete**
- **conf security login modify**
- **conf security login new**
- **conf security login password** (to modify passwords other than your own)
- **conf security login restore**
- **conf security login show**
- **conf security kerberos realm set**
- **conf security kerberos realm clear**
- **conf security kerberos host new**
- **conf security kerberos host delete**
- **conf security kerberos v5srvtab install**
- **conf security kerberos v5srvtab delete**
- all **conf security login securid** commands
- all **conf security ipaccess** commands
- all **conf security nsapfiltering** commands
- **conf snmp sets**
- all **debug** commands. The default is **admin**.

8.2.3 AMI Access Levels

The network administrator can configure four different levels of AMI access for userids: `serial`, `network`, `all`, and `none`. Each method is described in the following sections.

8.2.3.1 Serial Access

A person whose userid has a `serial` level of access is allowed to login to the switch only via the serial port.

8.2.3.2 Network Access

A person whose userid has a `network` level of access is allowed to login to the switch only via telnet.

8.2.3.3 All Access

A person whose userid has an `all` level of access is allowed to login to the switch via the serial port and via telnet.

8.2.3.4 No Access

A person whose userid has a `none` level of access is not allowed to log in to the switch at all.

8.2.4 Userid Password

The network administrator may also assign a password to a userid using the **`conf security login password`** command. (This command replaces the old **`oper password`** command.) When a user logged in with `user` privileges wants to modify the password, he or she must correctly enter the old password first before typing the new password. However, a user logged in with `admin` privileges can change any userid password without first entering the old local password, except their own. The maximum size is for a password is 512 characters. Any characters are allowed, except the colon (:) character.

8.2.5 Privilege Level for Unlisted Users

One of the login scenarios in Table 8.1 allows users who are not listed in the switch to login, provided that the SecurID server is accessible and provided that the SecurID server accepts the userid. The network administrator may assign a privilege level of `user` (access to only a certain subset of the AMI commands) or a privilege level of `admin` (access to all AMI commands) for these users via the **`conf security login upriv`** command.

8.3 IP Filtering

The IP filtering feature lets the network administrator limit access to the control port of the switch to prevent unauthorized access to the switch. The switch performs filtering on incoming IP packets by determining if there is a match between the packet's header source address and this table of authorized incoming IP addresses. If the addresses match, the packets are accepted, provided that they meet the requirements set up by the other IP filtering flags; otherwise, they are rejected. Statistics are kept of the number of rejected IP packets and about the last IP packet that was rejected.

8.3.1 Authorized IP Address Table

Using the **`conf security ipaccess accept`** command, the administrator can create an entry in a table of authorized IP addresses from which IP packets will be accepted. When the administrator creates an IP address entry in the table, he or she can apply a mask to specify a wildcard range of allowable addresses. For example, an IP address of 163.26.54.6 with a mask of 255.255.255.255 means only address 163.26.54.6 is accepted against that entry. However, an IP address of 163.26.54.6 with a mask of 255.255.0.0 means addresses 163.26.*.* are accepted against that entry. The table can contain a maximum of 32 entries.

The administrator can also delete entries from the table and display the contents of the table. The table will persist across a reboot. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information.)

CAUTION ⚠

When the authorized IP address table is empty, all addresses are accepted. This is the default state. Therefore, it is recommended that at least one address be entered into the table. Otherwise, anyone may access the switch via the control port. The address you enter must be the address of the machine you are using. Otherwise, you will lock yourself out of the switch.

8.3.2 IP Filtering Flags

There are three IP filtering flags that can be configured to limit IP access in other ways: `ssr`, `lsr`, and `all`. These flags are set to `allow` or `disallow` IP packets using the **`conf security ipaccess ssr`**, **`lsr`**, and **`all`** commands. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about these commands.)

8.3.2.1 Strict Source Routing Flag

Strict source routing is a setting within each IP packet. When enabled, it means that the sender is providing the specific path that the IP packet must use. Because source routing allows anyone to send arbitrary packets anywhere, it may be desirable to reject these types of packets for security reasons.

If the `ssr` flag is set to `allow`, all incoming IP packets that are strict source routed are accepted, provided that they match an IP address in the table of authorized addresses. If the `ssr` flag is set to `disallow`, all incoming IP packets that are strict source routed are rejected, even if they match an IP address in the table of authorized addresses.

8.3.2.2 Loose Source Routing Flag

Loose source routing is a setting within each IP packet. When enabled, it means that the sender is providing a specific list of IP addresses that the packet must pass through, but the packet may also pass through other routers between any two addresses in the list. Because source routing allows anyone to send arbitrary packets anywhere, it may be desirable to reject these types of packets for security reasons.

If the `lsr` flag is set to `allow`, all incoming IP packets that are loose source routed are accepted, provided that they match an IP address in the table of authorized addresses. If the `lsr` flag is set to `disallow`, all incoming IP packets that are loose source routed are rejected, even if they match an IP address in the table of authorized addresses.

8.3.2.3 All Flag

If the `all` flag is set to `allow`, all incoming IP packets are accepted, provided that they match an IP address in the table of authorized addresses. Allowing all is the default setting. If the `all` flag is set to `disallow`, all incoming IP packets are rejected, even if they match an IP address in the table of authorized addresses.

8.3.3 IP Access Statistics

The network administrator can display the total number of IP packets that have been filtered since the switch was rebooted using the **`stat ipaccess`** command. This command also displays information about the last IP packet that was dropped. This information includes the following:

- the reason that the last IP packet was dropped
- the system time at which the last IP packet was dropped
- the name of the interface on which the last dropped IP packet was received
- the IP address contained in the source field of the header of the last IP packet that was dropped

8.4 NSAP Filtering

This feature provides a mechanism for filtering calls based on a combination of the calling (source) and called (destination) addresses, as well as the incoming and outgoing UNIs.

Each UNI may have one address filter for incoming call setups, and one for outgoing call setups. If a call setup is routed from incoming UNI A to outgoing UNI B, A's incoming call filter and B's outgoing call filter are applied. A call setup message must be accepted by both filters if both are present.

NOTE

SPVCs, SPVPs, PVPs, and PVCs are not supported by NSAP filtering.

This feature also has a filter lookup mechanism and it provides statistics about calls that were filtered and information about the last call that was rejected by the filtering process.

8.4.1 Filters and Templates

Each filter is composed of an ordered set of templates. A template consists of the following:

- an incoming UNI (port/VPI)
- a source NSAP address/mask
- an outgoing UNI (port/VPI)
- a destination NSAP/Mask
- an action to either accept or reject the UNI or NSAP address

Masks may be wildcarded using a 0. Ports, VPIs, and NSAP addresses may be wildcarded using an asterisk (*). If * is specified for the port, then any port is accepted by the filter. If * is specified for the VPI, then any VPI is accepted by the filter. If * is specified for the NSAP address, or if 0 is specified for the mask, then any NSAP address is accepted by the filter.

Templates within a filter are applied in the order in which they appear in the filter, not by maximum prefix match. See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about how to create, modify, and delete filters and templates.

8.4.2 NSAP Filtering Lookup

A filter lookup mechanism is provided which allows you to enter components of a call setup message to test whether a call setup attempt with the supplied addresses and ports would be accepted or rejected by a specific filter. The switch returns an answer of `accepted` or `rejected` and the index number of the template that accepted or rejected the information entered. If the information does not match any of the existing templates, then an answer of `rejected` and `address unknown` is given. When this feature is used, none of the statistics are incremented and no traps are sent.

8.4.3 NSAP Filtering Statistics

To display statistics for NSAP filtering, use the **`stat nsapfilter`** command. See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about this command. NSAP filtering statistics and traps include:

- counts of accepted calls
- calls rejected because of an explicit match with a reject template
- calls rejected because of no match with any template

A trap is sent when rejected calls occur more frequently than the user-specified limit.

CHAPTER 9 Configuring Timing

This chapter describes how to set up timing on a FORE ATM Switch. Topics covered include:

- **Section 9.1 -** Overview
- **Section 9.2 -** Timing Modes
- **Section 9.3 -** Switchclock
- **Section 9.4 -** Port Level Timing
- **Section 9.5 -** Timing Configuration Examples

9.1 Overview

When running voice or video over an ATM network, the equipment on the network should be timed so that all of the equipment is transmitting and receiving data in a synchronized manner. Even though ATM is asynchronous, this only refers to the cell-level transmission that occurs in the ATM Layer. The physical layer underlying the cells is synchronous. Therefore, a time reference signal must be distributed to every element in the network to establish one cohesive entity. This time reference on FORE ATM Switches is called a *switchclock*.

9.2 Timing Modes

There are two timing modes available on FORE ATM Switches: TCM and switch. TCM mode means that all network modules that support distributed timing import their clock source from the Timing Control Module (TCM).

NOTE

You must have a TCM installed in your switch in order to use this timing mode. See the *CEC-Plus User's Manual* for information about configuring timing with a TCM. All examples in this chapter assume that you do not have a TCM installed.

Switch mode means that all network modules that support distributed timing import their clock source from the designated switchclock port. This is the default mode for any switch that does not have a TCM installed. See Section 9.3 for more information about this mode.

9.3 Switchclock

If a TCM is not installed in the switch, all of the ports within a fabric use the switchclock as their timing reference. The switchclock can be any port that is able to recover a clock (i.e., the network module supports distributed timing). (To see if the network module supports distributed timing, use the command **`conf module show`** and look for `yes` under the `Timing` field.) In an *ASX*-1000, *ASX*-1200, or a *TNX*-1100, the port may be from another fabric in the same chassis.

When the switchclock is set to any of the ports within the same fabric, that port's clock is exported to all of the network modules within the same fabric. On an *ASX*-1000, *ASX*-1200, or a *TNX*-1100 only, the switchclock is also exported to the network modules in all of the other fabrics in this chassis. You can then configure each of the other fabrics to use that clock. In this way, all of the network modules in all of the fabrics will use the same timing source.

9.3.1 Failover of the Switchclock

You can configure a *primary* switchclock and a *secondary* switchclock. If the primary clock is valid, it is used. If the primary switchclock fails, the secondary clock is used. When the primary clock returns, it is used again as the clock source.

If both the primary and secondary values fail, the switch fabric uses the crystal of the first available timing network module as the switchclock, going from A to D. For example, if network module A supports distributed timing, then the crystal from A is used as the switchclock. As another example, if network module A is not installed, and B and C do not support distributed timing, but D does support distributed timing, then the crystal from D is used as the switchclock. If the crystal of the network module has to be used for timing support, other ports from other fabrics can use that particular network module crystal as their timing reference. The network module crystal is of Stratum 4 level accuracy.

Failover of the switchclock happens automatically when any of the following events occur:

- A carrier detect or carrier loss occurs on any port in this switch fabric.
- A network module is swapped in or out.
- A switch fabric is swapped in or out.

9.4 Port Level Timing

In addition to the switchclock, there is also a configurable parameter for each port called a *transmit* clock (*txclock*). Each port's txclock can be configured to use either the *network* clock or the *internal* clock.

- When the txclock is set to `network`, the clock that is recovered from the receive line of a port is used to drive the transmit line of that port.
- When the txclock is set to `internal`, the internal clock, or switchclock, is used to drive the transmit line of that port.

NOTE

The `Timing` field under **`conf port sonet show`** displays `N/A` for all series of OC-12 network modules because they always use internal timing.

9.5 Timing Configuration Examples

This section provides examples of how to set the switchclock on various types of switches. For more information about the specific timing commands, see Part 2 of the *AMI Configuration Commands Reference Manual*.

If you have a TCM installed in your switch, see the *CEC-Plus User's Manual* for information about configuring timing with a TCM. All examples in this chapter assume that you do not have a TCM installed.

9.5.1 Configuring Timing on an *ASX*-200BX, *TNX*-210, or *ASX*-200WG

This example assumes that you are going to use 1A1 as your primary clock and 1B1 as your secondary clock. Use the following ATM Management Interface (AMI) commands to set your primary and secondary switchclocks:

```
conf timing switchclock primary 1a1

conf timing switchclock secondary 1b1
```

9.5.2 Configuring Timing on an *ASX*-1000, *ASX*-1200, or *TNX*-1100 (Single Timing Domain)

This example assumes that you are going to use 3A1 as your primary clock and 3B1 as your secondary clock. The only limitation on an *ASX*-1000, *ASX*-1200, or a *TNX*-1100 is that you must configure both the primary and secondary clocks to be on the same fabric. On all fabrics installed in the switch, use the following AMI commands to set your primary and secondary clocks:

```
conf timing switchclock primary 3a1

conf timing switchclock secondary 3b1
```

9.5.3 Configuring Timing on an *ASX*-1000, *ASX*-1200, or *TNX*-1100 (Multiple Timing Domains)

This example assumes that you are going to use 1C1 as your primary clock and 1D1 as your secondary clock for fabrics 1 and 2 and that you are going to use 3A1 as your primary clock and 3B1 as your secondary clock for fabrics 3 and 4. The only limitation on an *ASX*-1000, *ASX*-1200, or a *TNX*-1100 is that you must configure both the primary and secondary export clocks to be on the same fabric.

1. On fabrics 1 and 2, use the following AMI commands to set your primary and secondary clocks:

```
conf timing switchclock primary 1c1

conf timing switchclock secondary 1d1
```

2. On fabrics 3 and 4, use the following AMI commands to set your primary and secondary clocks:

```
conf timing switchclock primary 3a1

conf timing switchclock secondary 3b1
```

9.5.4 Configuring Timing on a *ForeRunnerLE* 155 or LE 25

The ports in interface groups A, B, C, and the Port Expansion Module (PEM) (interface D) use the crystal on the fabric as their timing source by default. Only the PEM can recover a clock from any of its ports and export it to the other interfaces. If the PEM is installed, the primary and secondary clocks be recovered from one of the PEM ports.

This example assumes that you are going to use 1D1 as your primary clock and 1D2 as your secondary clock. Use the following AMI commands to set your primary and secondary clocks:

```
conf timing switchclock primary 1d1

conf timing switchclock secondary 1d2
```

9.5.5 Configuring Timing on an *ASX*-4000

This example assumes that you are going to use 1A1 as your primary clock and 1B1 as your secondary clock. Use the following AMI commands to set your primary and secondary switchclocks:

```
conf timing switchclock primary 1a1

conf timing switchclock secondary 1b1
```

9.5.6 Configuring Timing on an *ESX*-3000

The ports in the interface groups on the *ESX*-3000 ATM port cards use their own crystal as their timing source by default. Only the first port on each port card (e.g., 3C1, 5A1, etc.) can recover a clock from any of its ports and export it to the other interface groups. The primary <u>or</u> the secondary clock can be recovered from the first port on any port card. Only one timing source can be configured per port card.

This example assumes that you are going to use 3B1 as your primary clock and 1C1 as your secondary clock. Use the following AMI commands to set your primary and secondary clocks:

```
conf timing switchclock primary 3b1

conf timing switchclock secondary 1c1
```

In the example above, the primary timing source is set to 3B1. The switch exports the timing source from the ATM port card in slot 3 to the entire chassis.

APPENDIX A Configuring SNMP

The switch control software for the FORE ATM switches includes an SNMP agent. The SNMP agent enables the remote monitoring and configuration of these switches.

A.1 SNMP Indexing

For a description of the SNMP indexing scheme used on FORE switches, please see the *ATM Switch Diagnostics and Troubleshooting Manual*.

A.2 SNMP Traps

SNMP traps are used to update the state of the network automatically to remote network management hosts. The SNMP agent on the switch supports several SNMP traps.

The traps generated by the switch's SNMP agent can be sent to as many destinations as needed. These destinations are configurable via the ATM Management Interface (AMI). Each destination must be an IP address of a network management host. The network management host specified for a trap destination can be any host with which the switch has connectivity. This means that the host does not have to be a directly connected ATM host. It can be on any attached network. Table A.1 describes the supported traps.

Table A.1 - SNMP Traps Supported on the FORE Switches

Trap Number	Trap Name	Description
0	asxSwLinkDown	An asxSwLinkDown trap signifies that the sending protocol entity recognizes a failure in one of the ATM Switch links that is connected to another switch.
1	asxSwLinkUp	An asxSwLinkUp trap signifies that the sending protocol entity recognizes that one of the ATM Switch links that is connected to another switch has come up.
2	asxHostLinkDown	An asxHostLinkDown trap signifies that the sending protocol entity recognizes a failure in one of ATM Switch links that is connected to a host.
3	asxHostLinkUp	An asxHostLinkUp trap signifies that the sending protocol entity recognizes that one of the ATM Switch links that is connected to a host has come up.
4	asxNetModuleDown	An asxNetModuleDown trap signifies that the sending protocol entity recognizes a failure in one of the ATM Switch network modules, that is identified by the board and the module numbers. This is probably caused by a hot-swap of a network module.
5	asxNetModuleUp	An asxNetModuleUp trap signifies that the sending protocol entity recognizes a new operational ATM Switch network module, that is identified by the board and the module numbers. This is probably caused by a hot-swap of a network module.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
6	asxPsInputDown	This trap alerts that one ATM switch power supply failed due to failure in the input voltage. The power supply that failed is identified by the power supply index. Note that an input voltage may be out of specification and may not cause a power supply failure if high loads are not applied.
7	asxPsInputUp	This trap alerts that one ATM switch power supply that had an input failure is up. The power supply that is back up is identified by the power supply index.
9	asxPsOutputDown	This trap alerts that one ATM switch power supply output or the power supply was physically removed. The power supply that failed is identified by the power supply index.
10	asxPsOutputUp	This trap alerts that one ATM switch power supply that had an output failure or was removed is now up. The power supply that is back up is identified by the power supply index.
22	asxFanBankDown	This trap alerts that one ATM switch fan bank failed. The fan bank that failed is identified by the fan bank index.
23	asxFanBankUp	This trap alerts that one ATM switch fan bank is up. The fan bank that is back up is identified by the fan bank index.
28	asxLinkDown	This trap alerts that the link that is identified by {hwPortBoard, hwPortModule, hwPortNumber} was configured up but lost its carrier (or the framing bit) and is currently down.
29	asxLinkUp	This trap alerts that the link that is identified by {hwPortBoard, hwPortModule, hwPortNumber} is back up.
30	asxSpansDown	This trap alerts that the SPANS signalling on the link that is identified by the sigPathPort and sigPathVPI failed.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
31	asxSpansUp	This trap alerts that the SPANS signalling on the link that is identified by the sigPathPort and sigPathVPI is up.
32	asxTempSensorOverTemp	This trap alerts that one of the temperature sensors indicates over temperature. The temperature sensor is identified by the temperature sensor index.
33	asxTempSensorRegularTemp	This trap alerts that one of the temperature sensors indicates regular temperature. The temperature sensor is identified by the temperature sensor index.
34	asxFabricTemperature OverTemp	This trap alerts that one of the temperature sensors indicates over temperature. The temperature sensor is identified by the temperature sensor index.
35	asxFabricTemperature RegularTemp	This trap alerts that one of the temperature sensors indicates regular temperature. The temperature sensor is identified by the temperature sensor index.
36	asxSonetLOSon	This trap indicates that the specified SONET port is experiencing Loss Of Signal. Bellcore Document TA-NWT-000253 Section 6.3.1.1.1 states that, “A SONET NE shall declare a LOS failure when the LOS defect persists for 2.5 (± .5) seconds, or when a LOS defect is present and the criteria for LOF failure declaration have been met.”
37	asxSonetLOSoff	This trap indicates that the LOS condition identified by trap asxSonetLOSon has been cleared.
38	asxSonetPathLabelOn	This trap indicates that the specified SONET port is receiving and errored C2 Path Label byte. Reference Bellcore Document TA-NWT-000253 Section 3.3.2.3 and 6.3.1.1.8 the Path Label (C2) byte should have the value 0x13.
39	asxSonetPathLabelOff	This trap indicates that the Errored Path Label (C2) byte error condition signalled by the asxSonetPathLabelOn trap has been cleared.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
40	asxSonetLineAISon	This trap indicates that the specified SONET port is receiving a Line level Alarm Indication Signal from the far-end equipment.
41	asxSonetLineAISoff	This trap indicates that the Line AIS error condition signalled by the asxSonetLineAISon trap has been cleared.
46	asxDS3PLCPYellowDetected	This trap indicates that the specified DS3 port has detected incoming Yellow Alarm.
47	asxDS3PLCPYellowCleared	This trap indicates that the specified DS3 port has detected clearance of incoming Yellow Alarm.
48	asxDS3PLCPLOFDetected	This trap indicates that the specified DS3 port has detected incoming LOF Alarm.
49	asxDS3PLCPLOFCleared	This trap indicates that the specified DS3 port has detected clearance of incoming LOF Alarm.
50	asxDS3LOFDetected	This trap indicates that Loss Of Frame(LOF) is detected on the incoming signal.
51	asxDS3LOFCleared	This trap indicates that Loss Of Frame is cleared on the incoming signal.
52	asxDS3AISDetected	This trap indicates that AIS Alarm is detected on the incoming signal.
53	asxDS3AISCleared	This trap indicates that AIS Alarm is cleared on the incoming signal.
60	asxDS1PLCPYellowDetected	This trap indicates that the specified DS1 port has detected an incoming Yellow Alarm.
61	asxDS1PLCPYellowCleared	This trap indicates that the specified DS1 port has detected clearance of an incoming Yellow Alarm.
62	asxDS1PLCPLOFDetected	This trap indicates that the specified DS1 port has detected an incoming LOF Alarm.
63	asxDS1PLCPLOFCleared	This trap indicates that the specified DS1 port has detected clearance of an incoming LOF Alarm.
64	asxDS1YellowDetected	This trap indicates that Yellow Alarm is detected on the incoming signal.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
65	asxDS1YellowCleared	This trap indicates that Yellow Alarm is cleared on the incoming signal.
66	asxDS1AISDetected	This trap indicates that AIS Alarm is detected on the incoming signal.
67	asxDS1AISCleared	This trap indicates that AIS Alarm is cleared on the incoming signal.
68	asxDS1LOSDetected	This trap indicates that LOS Alarm is detected on the incoming signal.
69	asxDS1LOSCleared	This trap indicates that LOS Alarm is cleared on the incoming signal.
70	asxDS1LOFDetected	This trap indicates that LOF Alarm is detected on the incoming signal.
71	asxDS1LOFCleared	This trap indicates that LOF Alarm is cleared on the incoming signal.
74	asxDS3FERFDetected	This trap indicates that FERF Alarm is detected on the incoming signal.
75	asxDS3FERFCleared	This trap indicates that FERF Alarm is cleared on the incoming signal.
78	asxE3YellowDetected	This trap indicates that the Yellow Alarm is being detected on the incoming signal.
79	asxE3YellowCleared	This trap indicates that Yellow alarm has cleared on the incoming signal.
80	asxE3OOFDetected	This trap indicates that Out Of Frame (OOF) is detected on the incoming signal.
81	asxE3OOFCleared	This trap indicates that Loss Of Frame is cleared on the incoming signal.
82	asxE3AtmLCDDetected	This trap indicates that the specified E3 port is experiencing Loss of Cell Delineation (LCD). An LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
83	asxE3AtmLCDCleared	This trap indicates that the LCD failure identified by trap asxE3AtmLCDDetected has been cleared. An LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
86	asxE3PLCPYellowDetected	This trap indicates that the specified E3 port has detected incoming PLCP Yellow Alarm.
87	asxE3PLCPYellowCleared	This trap indicates that the specified E3 port has detected clearance of incoming PLCP Yellow Alarm.
90	asxE1YellowDetected	This trap indicates that the Yellow Alarm is being detected on the incoming signal.
91	asxE1YellowCleared	This trap indicates that Yellow alarm has cleared on the incoming signal.
92	asxE1LOFDetected	This trap indicates that LOF is being detected on the incoming signal.
93	asxE1LOFCleared	This trap indicates that LOF is cleared on the incoming signal.
96	asxE1PLCPYellowDetected	This trap indicates that the specified E1 port has detected incoming PLCP Yellow Alarm.
97	asxE1PLCPYellowCleared	This trap indicates that the specified E1 port has detected clearance of incoming PLCP Yellow Alarm.
98	asxE1PLCPLOFDetected	This trap indicates that the specified E1 port has detected incoming PLCP LOF Alarm.
99	asxE1PLCPLOFCleared	This trap indicates that incoming PLCP LOF alarm has been cleared on the specified E1 port.
100	asxE1LOSDetected	This trap indicates that the specified E1 port has detected incoming LOS Alarm.
101	asxE1LOSCleared	This trap indicates that incoming LOS alarm has been cleared on the specified E1 port.
102	asxE1AISDetected	This trap indicates that the specified E1 port has detected incoming AIS Alarm.
103	asxE1AISCleared	This trap indicates that incoming AIS alarm has been cleared on the specified E1 port.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
104	asxE3AISDetected	This trap indicates that the specified E3 port has detected incoming AIS Alarm.
105	asxE3AISCleared	This trap indicates that incoming AIS alarm has been cleared on the specified E3 port.
106	asxE3LOSDetected	This trap indicates that the specified E3 port has detected incoming LOS Alarm.
107	asxE3LOSCleared	This trap indicates that incoming LOS alarm has been cleared on the specified E3 port.
108	asxE3PLCPLOFDetected	This trap indicates that the specified E3 port has detected incoming PLCP LOF Alarm.
109	asxE3PLCPLOFCleared	This trap indicates that incoming PLCP LOF alarm has been cleared on the specified E3 port.
112	asxJ2YellowDetected	This trap indicates that Yellow Alarm is detected on the incoming signal.
113	asxJ2YellowCleared	This trap indicates that Yellow Alarm is cleared on the incoming signal.
114	asxJ2AISDetected	This trap indicates that AIS Alarm is detected on the incoming signal.
115	asxJ2AISCleared	This trap indicates that AIS Alarm is cleared on the incoming signal.
116	asxJ2LOSDetected	This trap indicates that LOS Alarm is detected on the incoming signal.
117	asxJ2LOSCleared	This trap indicates that LOS Alarm is cleared on the incoming signal.
118	asxJ2LOFDetected	This trap indicates that LOF Alarm is detected on the incoming signal.
119	asxJ2LOFCleared	This trap indicates that LOF Alarm is cleared on the incoming signal.
120	asxDS3LOSDetected	This trap indicates that the specified DS3 port has detected incoming LOS Alarm.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
121	asxDS3LOSCleared	This trap indicates that the incoming LOS Alarm has been cleared on the specified DS3 port.
130	asxSonetLOFDetected	This trap indicates that the specified SONET port is experiencing Loss Of Frame (LOF) failure. An LOF failure is declared when the LOF defect persists for a period of 2.5 +/- 0.5 seconds, except when an LOS defect or failure is present.
131	asxSonetLOFCleared	This trap indicates that the LOF failure identified by trap asxSonetLOFDetected has been cleared. The LOF failure is cleared when the LOS failure is declared, or when the LOF defect is absent for 10 +/- 0.5 seconds.
132	asxSonetLineRDIDetected	This trap indicates that the specified SONET port is experiencing Line Remote Defect Indication (LRDI). A Line RDI failure is declared when the incoming Line RDI defects lasts for 2.5 +/- 0.5 seconds.
133	asxSonetLineRDICleared	This trap indicates that the Line RDI failure identified by trap asxSonetLineRDIDetected has been cleared. The Line RDI failure is cleared when no Line RDI defects are detected for 10 +/- 0.5 seconds.
134	asxSonetPathAISDetected	This trap indicates that the specified SONET port is experiencing Path Alarm Indication Signal (PAIS). A Path AIS failure is declared when the Path AIS defect persists for 2.5 +/- 0.5 seconds.
135	asxSonetPathAISCleared	This trap indicates that the Path AIS failure identified by trap asxSonetPathAISDetected has been cleared. A PAIS failure is cleared when the PAIS defect is absent for 10 +/- 0.5 seconds.
136	asxSonetPathLOPDetected	This trap indicates that the specified SONET port is experiencing Loss Of Pointer (LOP). A LOP failure is declared when the LOP defect persists for a period of 2.5 +/- 0.5 seconds.
137	asxSonetPathLOPCleared	This trap indicates that the LOP failure identified by trap asxSonetLOPDetected has been cleared. A LOP failure is cleared when the LOP defect is absent for 10 +/- 0.5 seconds.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
138	asxSonetPathUNEQDetected	This trap indicates that the specified SONET port is experiencing unequipped (UNEQ). A UNEQ failure is declared when the UNEQ defect persists for a period of 2.5 +/- 0.5 seconds.
139	asxSonetPathUNEQCleared	This trap indicates that the UNEQ failure identified by trap asxSonetUNEQDetected has been cleared. A UNEQ failure is cleared when the UNEQ defect is absent for 10 +/- 0.5 seconds.
140	asxSonetPathRDIDetected	This trap indicates that the specified SONET port is experiencing Path Remote Defect Indication (PRDI). A Path RDI failure is declared when the incoming Path RDI defects lasts for 2.5 +/- 0.5 seconds.
141	asxSonetPathRDICleared	This trap indicates that the Path RDI failure identified by trap asxSonetPathRDIDetected has been cleared. The Path RDI failure is cleared when no Path RDI defects are detected for 10 +/- 0.5 seconds.
142	asxSonetAtmLCDDetected	This trap indicates that the specified SONET port is experiencing Loss of Cell Delineation (LCD). A LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.
143	asxSonetAtmLCDCleared	This trap indicates that the LCD failure identified by trap asxSonetAtmLCDDetected has been cleared. A LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
160	asxDS3IdleDetected	This trap indicates that an Idle Maintenance Signal (IDLE) is detected on the incoming signal.
161	asxDS3IdleCleared	This trap indicates that an Idle Maintenance Signal (IDLE) is cleared on the incoming signal.
162	asxDS3AtmLCDDetected	This trap indicates that the specified DS3 port is experiencing Loss of Cell Delineation (LCD). An LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
163	asxDS3AtmLCDCleared	This trap indicates that the LCD failure identified by trap asxDS3AtmLCDDetected has been cleared. An LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
176	asxDS1PRBSDetected	This trap indicates that PRBS pattern is detected on the incoming signal.
177	asxDS1PRBSCleared	This trap indicates that PRBS pattern is cleared on the incoming signal.
178	asxDS1AtmLCDDetected	This trap indicates that the specified DS1 port is experiencing Loss of Cell Delineation (LCD). An LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.
179	asxDS1AtmLCDCleared	This trap indicates that the LCD failure identified by trap asxDS1AtmLCDDetected has been cleared. An LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
192	asxE3TrailChangeDetected	This trap indicates that a change in the trail trace message was detected on the incoming signal.
208	asxE1AtmLCDDetected	This trap indicates that the specified E1 port is experiencing Loss of Cell Delineation (LCD). An LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.
209	asxE1AtmLCDCleared	This trap indicates that the LCD failure identified by trap asxE1AtmLCDDetected has been cleared. An LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
224	asxJ2RLOCDetected	This trap indicates that Receive Loss of Clock (RLOC) is detected on the incoming signal.
225	asxJ2RLOCCleared	This trap indicates that Receive Loss of Clock (RLOC) is cleared on the incoming signal.
226	asxJ2HBERDetected	This trap indicates that High Bit Error Rate (HBER) is detected on the incoming signal.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
227	asxJ2HBERCleared	This trap indicates that High Bit Error Rate (HBER) is cleared on the incoming signal.
228	asxJ2PAISDetected	This trap indicates that Payload Alarm Indication Signal (PAIS) is detected on the incoming signal.
229	asxJ2PAISCleared	This trap indicates that Payload Alarm Indication Signal (PAIS) is cleared on the incoming signal.
230	asxJ2AtmLCDDetected	This trap indicates that the specified J2 port is experiencing Loss of Cell Delineation (LCD). An LCD failure is declared when the LCD defect persists for a period of 2.5 +/- 0.5 seconds.
231	asxJ2AtmLCDCleared	This trap indicates that the LCD failure identified by trap asxJ2AtmLCDDetected has been cleared. An LCD failure is cleared when the LCD defect is absent for 10 +/- 0.5 seconds.
232	asxJ2TLOCDetected	This trap indicates that Transmit Loss of Clock (TLOC) is detected.
233	asxJ2TLOCCleared	This trap indicates that Transmit Loss of Clock (TLOC) is cleared.
250	asxTP25LOSDetected	This trap indicates that LOS Alarm is detected on the incoming signal.
251	asxTP25LOSCleared	This trap indicates that LOS Alarm is cleared on the incoming signal.
1024	asxOutputQueueCongested	This trap indicates that the output queue for the given priority has exceeded its dedicated length, and has begun overflowing into the shared buffer space on the network module.
1025	asxOutputQueueCellLoss	This trap indicates that the output queue for the given priority has overflowed and cells have been dropped.
1026	asxExtendedModeViolation	This trap indicates that a series A or B network module was inserted into a switch board running in extended mode.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
1027	asxNonextendedMode-Warning	This trap indicates that a series C or greater network module was inserted into a switch board running in non-extended mode.
1028	q2931AVRejectTrap	This trap is generated whenever any UNI3.x with AddressValidation enabled rejects a Setup Request call more than q2931AVRejectTrapThreshold times in any given q2931AVRejectTrapPeriod.
1029	crConfMemoryOflow	This trap is generated when the allocated call record memory (as indicated by crMemoryAllocated) is exceeded.
1030	crXfrPrimaryXfrFailed	This trap is generated when the call record transfer to the primary host (as indicated by crXfrPrimaryUrl) fails.
1031	crXfrSecondaryXfrFailed	This trap is generated when the call record transfer to the secondary host (as indicated by crXfrSecondaryUrl) fails.
1032	crConfMemAllocFail	This trap is generated when Callrecord functionality is unable to allocate memory as specified by crMemoryAllocated. This can happen when the crConfAdminStatus changes state from “off” or when the switch reboots when Callrecords is configured “on”.
1033	crGeneralFailure	This trap is generated when any of the callrecord related functionality fails for any reason. One example would be when the Callrecord Module fails to schedule an interval timer.
1034	asxDualScpSyncFailure	This trap indicates that automatic CDB synchronization is disabled due to failures.
1035	asxDualScpSwitchOver	This trap indicates that the backup SCP has taken control of the switch.
1036	asxDualScpHotSwap	This trap indicates that an SCP hotswap insertion or removal has occurred.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
1037	asxVPAISDetected	This trap indicates that the Alarm Indication Signal (AIS) is detected on the incoming (terminating) virtual path. This trap is generated once when the virtual path is declared to be in the active AIS state.
1038	asxVPAISCleared	This trap indicates that the Alarm Indication Signal (AIS) has been removed from the incoming (terminating) virtual path. This trap is generated once when the virtual path is declared to be in the inactive AIS state.
1039	asxVPRDIDetected	This trap indicates that the Remote Defect Indication (RDI) is detected on the incoming (terminating) virtual path. This trap is generated once when the virtual path is declared to be in the active RDI state.
1040	asxVPRDICleared	This trap indicates that the Remote Defect Indication (RDI) has been removed from the incoming (terminating) virtual path. This trap is generated once when the virtual path is declared to be in the inactive RDI state.
1041	asxNonextendedModeViolation	This trap indicates that a Series D network module was inserted into a switch board running in non-extended mode. Multicast will not work on the Series D module without removing all Series B modules and rebooting the switch.
1042	asxUnsupportedNetworkModule	This trap indicates that a unsupported network module was inserted into a switch.
1049	asxIpFilterViolation	This trap occurs when an incoming IP packet is unauthorized to enter the switch control port and has been dropped.
1053	q2931AFRejectKnown	This trap is generated whenever any q2931 UNI with Address Filtering enabled rejects a Setup request because the request matched a template with the action “reject.” The variables sent in the trap identify the source and destination UNI for the call.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
1054	q2931AFRejectUnknown	This trap is generated whenever any q2931 UNI with Address Filtering enabled rejects a Setup request because the address matched no template. The variables sent in the trap identify the source and destination UNI for the call.
1061	q2931CreationFailure	This trap is generated whenever a switch fails to create a UNI. This is most likely due to a resource limitation on the switch.
1068	asxPsCurrentDown	This trap alerts that one ATM switch power supply had a current failure. The power supply that failed is identified by the power supply index.
1069	asxPsCurrentUp	This trap alerts that one ATM switch power supply that had a current failure is now up. The power supply that is back up is identified by the power supply index.
1070	asxPs5VoltDown	This trap alerts that one ATM switch power supply had a +5V failure. The power supply that failed is identified by the power supply index.
1071	asxPs5VoltUp	This trap alerts that one ATM switch power supply that had a +5V failure is now up. The power supply that is back up is identified by the power supply index.
1072	asxSwitchLoginDetected	An asxSwitchLoginDetected trap signifies that a user logged in on the switch.
1073	asxSwitchLoginFailed	An asxSwitchLoginFailed trap signifies that a user's attempt to log in on the switch failed.
1074	pnniTdbGuardbandResrv-Fail	This trap is generated when the guardband memory reserve for any of the PNNI TDB (Topology Database) related functionality like creation, modification, and deletion on objects like node, PTSE, flags, internal prefixes, external prefixes, etc. fails. The switch is low on memory.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
1075	pnniTdbInconsistentState	This trap is generated when the PNNI TDB (Topology Database) is in an unrecoverable error condition due to MALLOC and other TDB related failures. When this happens, the associated logical node is shut down and this trap is sent. The switch has to be rebooted to bring this PNNI logical node up again.
1077	asxShmem2OutputQueue Congested	This trap indicates that the output queue for the given priority has exceeded its dedicated length, and has begun overflowing into the shared buffer space on the network module.
1078	asxShmem2OutputQueue CellLoss	This trap indicates that the output queue for the given priority has overflowed and cells have been dropped.
1080	fabricLvl3Lookup	This trap is generated when a fabric detects excessive level 3 lookup errors. The level 3 lookup count is read every second. If the lookup count is non-null for 5 consecutive seconds, then this trap is generated. If the errors continue, this trap is generated continuously every 5 seconds. This trap can be caused by: a misconfigured PVC; bad hardware.
1081	fabricCorrectedLookup	This trap is generated when the SCP detects inconsistent information in the HDCOMP lookup. The switch has corrected the bad information in the HDCOMP lookup ASIC's data structures.
1090	spvcRerouteInitiated	This trap is sent when a SPVC is being re-routed due to a better path being found.
1091	asxQ2931Down	This trap is generated whenever UNI signalling goes down.
1092	asxQ2931Up	This trap is generated whenever UNI signalling goes up.
1093	asxFabricDown	An asxFabricDown trap signifies that the sending protocol entity recognizes a failure in one of the *ASX*-4000 ATM Switch fabrics, that is identified by the board number. This is probably caused by a hot-swap of a fabric module.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
1094	asxFabricUp	An asxFabricUp trap signifies that the sending protocol entity recognizes a new operational ATM *ASX*-4000 Switch fabric, that is identified by the board number. This is probably caused by a hot-swap of a fabric module.
2000	frf8PVCStatus	This trap indicates when an interworking PVC has experienced an alarmed condition, either on the ATM network side or Frame Relay side. It is also generated when the PVC alarmed condition is cleared. It carries the operational status of the PVC by the frf8ConnOperStatus, as well as the reason why exiting, entering, or changing the alarmed state frf8ConnPVCAlarmReason. If the interworking PVC changes its status by an AdminStatus command (active/inactive/testing), causing the frf8ConnOperStatus to change (up/down), this trap is generated with frf8ConnPVCAlarmReason set to no defect. This trap gives the support to manage defects that might occur in either the Frame Relay network or the ATM network, even if the PVC Operational Status is already down.
2001	ifLinkDown	This trap indicates that the SNMP agent has detected that the ifOperStatus of this interface is about to transition into the down state.
2002	ifLinkUp	This trap indicates that the SNMP agent has detected that the ifOperStatus of this interface has transitioned out of the down state.
2003	framNakMsg	This trap is sent when a message from the SCP to the *FramePlus* network module does not succeed. The request that has failed is encoded within the trap as the `messageType`, and the reason for failure is encoded within the `errorCode`. See Table A.2 for the message types and the message requests. See Table A.3 for the error codes and their meanings.

Table A.1 - SNMP Traps Supported on the FORE Switches (Continued)

Trap Number	Trap Name	Description
2004	pnniSpvccDown	This trap is sent when an SPVC call is cleared or deleted. The pnniSpvcSrcDownReason indicates if the call was cleared due to a better route being found or a network failure. It also may indicate SPVC deletion. This trap is only sent if pnniSpvcTrapMode is all.
2005	pnniSpvccUp	This trap is send when an SPVC call is established. It identifies the SPVC and the old and new cost. If the call has not been up before, the pnniSpvcSrcOldRouteCost will be -1. This trap is only sent if pnniSpvcTrapMode is failure or all.
2006	pnniSpvccFail	This trap is sent when an SPVC call that is down fails to get connected on 16 consecutive attempts. This trap is only sent if pnniSpvcTrapMode is failure or all.
2007	pnniSpvpcDown	This trap is sent when an SPVP call is cleared. The pnniSpvcSrcDownReason indicates if the call was cleared due to a better route being found or a network failure. It also may indicate SPVPC deletion. This trap is only sent if pnniSpvpcTrapMode is all.
2008	pnniSpvpcUp	This trap is sent when an SPVP call is established. It identifies the SPVP and the old and new cost. If the call has not been up before, the pnniSpvcSrcOldRouteCost will be -1. This trap is only sent if pnniSpvpcTrapMode is all or failure.
2009	pnniSpvpcFail	This trap is sent when an SPVP call that is down fails to get connected on 16 consecutive attempts. This trap is only sent if pnniSpvpcTrapMode is all or failure.
2010	asxPortCardDown	An asxPortCardDown trap signifies that the sending protocol entity recognizes a failure in one of the ATM Switch port cards, that is identified by the port card index. This is probably caused by a hot-swap of the port card.
2011	asxPortCardUp	An asxPortCardUp trap signifies that the sending protocol entity recognizes a new operational ATM Switch port card, that is identified by the port card index. This is probably caused by a hot-swap of the port card.

Table A.2 defines the message types and the message requests for Trap 2003.

Table A.2 - Message Type Encodings for Trap 2003

MessageType	Message Request
0	Configure network module
1	Configure port
2	Send egress Fdl
4	Network module reset
5	Port reset
6	Configure a channel / service
7	Request channel / service configuration
8	Request port configuration
9	Request connection configuration
10	Channel / service reset
17	Configure LEDs
28	Request application revision
29	Create a FUNI service
30	Create a Frame Relay service
31	Request to change service admin status
32	Request to enable / disable service statistics
33	Request to enable / disable network module statistics
34	Request to enable / disable network module admin status
35	Request to enable / disable port admin status
36	Delete a service
37	Create a Frame Relay connection
38	Create a FUNI connection
40	Request to enable / disable connection admin status
41	Request to enable / disable service egress rate enforcement

Table A.2 - Message Type Encodings for Trap 2003 (Continued)

MessageType	Message Request
42	Request to enable / disable connection ingress rate enforcement
43	Delete a connection
47	Configure port to be timing source

Table A.3 defines the error codes for Trap 2003 and their meanings.

Table A.3 - Error Codes for Trap 2003

Error Code	Error Code Meaning
1	Invalid or out of range parameter
2	System disabled
3	Memory allocation failure
4	Timer allocation failure
5	Circuits on link exceeded
6	Invalid CIR configuration
7	No CCB for link
8	Unknown link
9	Unknown DLCI
0A	Link number out of range
0B	DLCI number out of range
0C	Link already exists
0D	Circuit on link already exists
0E	Link contains circuits
0F	Out of memory
10	Out of timers
11	Out of timers
12	Length error
13	Duplicate connection ID

Table A.3 - Error Codes for Trap 2003 (Continued)

Error Code	Error Code Meaning
14	Invalid parameter
15	Duplicate DLCI
16	Memory allocation failure
17	Service not created
18	Connection does not exist
19	Invalid Service
1A	Connection ID unavailable
1B	Invalid Connection ID
1C	Unexpected message
1D	Object already configured
21	Operation failed
22	Physical channel configuration failed
23	Physical connection configuration failed
24	Physical channel teardown failed
25	Physical connection teardown failed
26	Port configuration failed
27	EPD/PPD configuration failed
28	Unknown SDPM message received
29	Netmod not configured
2A	SDPM message size error
2B	Port not configured
2C	Allocation of buffer failed

A.2.1 Adding SNMP Trap Destinations

To create one or more SNMP trap destinations on a FORE switch, log in to AMI and open a session on the switch. Enter the following parameters:

```
configuration snmp trap destinations new <ipaddress>
```

The *<ipaddress>* variable indicates the IP address of the SNMP trap destination that is to be created. Repeat this for as many SNMP trap destinations as needed. Traps are active as soon as they are set.

A.2.2 Displaying SNMP Trap Destinations

To list all of the current SNMP trap destinations, log in to AMI and open a session on the switch. The SNMP traps supported by this switch are detailed in the FORE-Switch-MIB. Enter the following parameters:

```
configuration snmp trap destinations show
```

The switch responds with a list similar to the following:

```
Trap    Destination
1       192.88.243.18
2       198.29.16.14
3       198.29.16.18
```

If no trap destinations have been configured, then the following is displayed:

```
myswitch::configuration snmp trap destinations> show
No trap information is available
```

A.2.3 Removing SNMP Trap Destinations

To delete one or more SNMP trap destinations for a FORE switch, log in to AMI and open a session on the switch. Prior to deleting any trap that may need to be recreated later, as a precaution, a recommended practice is to list all trap destinations using AMI and either copy the screen or write down the destinations. To delete a trap, enter the following parameters:

```
configuration snmp trap destinations delete <trap>
```

The *<trap>* variable indicates the index number of the SNMP trap destination that is to be removed. Repeat this for as many SNMP trap destinations as needed.

NOTE

For more information about the SNMP trap destination commands, see Part 2 of the *AMI Configuration Commands Reference Manual*.

APPENDIX B Configuring Circuit Emulation Services

FORE Systems' Circuit Emulation Services (CES) Network Modules (NMCE-6/DS1A and NMCE-6/E1A) provide adaptation from time-division multiplexed (TDM) equipment (i.e., PBXs, WAN multiplexers, channel banks, video codecs, etc.) and traffic to ATM. Both modules provide structured and unstructured services, with a maximum of 127 connections supported on each module.

- All six ports of the DS1 CES Network Module may support fractional DS1 services (n x 56 Kbps/n x 64 Kbps) where 1 to 24 contiguous or non-contiguous DS0 channels are mapped to a single ATM VCC not to exceed 127 total connections.
- All six ports of the E1 CES Network Module may support fractional E1 services (n x 56 Kbps/n x 64 Kbps) where 1 to 31 contiguous or non-contiguous DS0 channels are mapped to a single ATM VCC not to exceed 127 total connections.

Structured services provide digital access and cross-connect system (DACS) connectivity where n x 64 Kbps and n x 56 Kbps digital signal level zero (DS0) channels are adapted to ATM cells and mapped to unique ATM virtual connections (VCCs).

Unstructured services provide support and maintenance of a single full bandwidth 1.544 Mbps (DS1) or 2.048 Mbps (E1) clear channel across a single ATM virtual connection.

Configurations of both the DS1 and the E1 version of the CES network module are detailed in this chapter. This chapter assumes the proper configuration of general switch parameters and of CES specific parameters, such as timing distribution/recovery, additional CES alarms, CES ports, etc.

B.1 Idle Channel Suppression

Idle channel suppression is a method used to detect active and inactive timeslots (data paths of size DS0) in a TDM trunk in an ATM network based on CES. An idle timeslot is a timeslot in a connection through which no useful data is transmitted; e.g., the end users may be silent in a voice connection. An active connection is one in which at least one timeslot is not idle, and an inactive connection is one in which all of the timeslots are idle.

Because the CES services are CBR services, when a CES connection is created, the switch allocates a fixed amount of bandwidth, depending on the number of timeslots used. When idle channel suppression is enabled on a CES connection, the CES network module transmits only the cells of active connections to the switch fabric. Thus, only a fraction of the bandwidth allocated for the connection is used. The egress network module detects that there is less than the contracted cell rate arriving on the CES connections, and re-allocates the extra bandwidth to other existing VBR, ABR, and UBR connections, but does not exceed their contracted cell rates. Additionally, the egress network module guarantees the constant bit rate for CES connections whenever they want to reclaim the bandwidth.

When CES SPVCs or PVCs are created, the virtual channels assigned for these connections at the intermediate switches are also assigned CBR service. The unused bandwidth in the intermediate connections are also reused by other bit rate services. Idle channel suppression works slightly differently for Channel Associated Signalling (CAS) connections than it does for basic connections.

B.1.1 CAS Connections

The timeslot state is determined according to CAS. When all of the timeslots on a given connection become idle, the connection becomes idle. A short period of time (the integration period) is waited, during which the idle patterns are observed, before the channel state is changed from active to idle. When a connection is idle, OAM F5 cells are used to maintain the connection. When one or more channels on a given connection become active, the connection becomes active. When an active connection is detected, the change from idle to active is immediate.

B.1.1.1 Idle Detection using Signalling Bits

This method can be used only for structured CAS connections and is a recommended method for voice connections. The majority of these applications use the AB bit idle state patterns. The CAS patterns supported are as follows: E&M, Loop Start FXS, Loop Start FX0, Ground Start FXS, and Ground Start FX0. These patterns can be user-to-network or network-to-network. The available options are as follows: em00, em01, fxolsuser01, fxolsuser11, fxolsnet00, fxolsnet01, fxslsuser00, fxslsuser01, fxslsnet01, fxslsnet11, fxsgsuser01, fxsgsnet10, fxsgsnet11, fxogsuser10, fxogsuser11, fxogsnet01, r210. The default pattern is em00. The default integration period is 1 second and is user-configurable from 500 μs to 2 seconds.

B.1.1.2 Active Detection

A connection is declared active when at least one timeslot becomes active. The active detection mechanism is not user-configurable. The AB bits are compared against the configured pattern for idle states, and if different, the connection is immediately declared active.

B.1.2 Basic Connections

For Basic (Common Channel Signalling (CCS)) connections, only one idle pattern is used to determine the connection state. When the data received from the line interface matches the idle pattern for a short period of time (the integration period), the connection becomes idle. When a connection is idle, OAM F5 cells are used to maintain the connection. When the first data cell that is received does not match the idle pattern, the connection immediately becomes active.

NOTE

Idle channel suppression is not supported on unstructured connections.

Structured basic connections require a partial fill size of at least 24 to support idle channel suppression.

B.1.2.1 Idle Detection using Pattern Matching

This method of idle detection is used only for basic (CCS) services. Depending on the applications, the idle pattern can vary. Additionally, different idle patterns can be used to identify, for example, an out-of-service channel.

In this implementation, a single idle pattern determines the idle status of a channel. The default idle pattern is FF.

An idle code pattern must be consistent during an integration period before it can be determined that a time slot has become idle. This eliminates the false detection of an idle state due to the natural occurrence of the specified idle data patterns in the connection’s normal data content from time to time. For basic connections, the default integration period is 1 second, and is user-configurable from 500 μs to 2 seconds.

B.1.2.2 Idle Detection using Mask Patterns

This method of idle detection is used only for basic (CCS) services. The idle patterns can be provided by the user in a mask format. The mask pattern is provided in a format, for example, F0 hex. This means that pattern matching is performed only on the four most significant bits. This method is useful, for example, in T1 robbed-bit signalling systems.

B.1.2.3 Active Detection

A connection is declared active when at least one timeslot becomes active. The active detection mechanism is not user-configurable. If at least one of the timeslots is not matched against the idle patterns configured, the connection is declared active. The integration period is not user-configurable and the transition from idle to active is immediate; i.e., when the first cell does not have idle patterns.

B.1.3 Idle Channel Suppression Configuration Examples

Various examples are provided in this section for configuring idle channel suppression.

NOTE

When idle channel suppression is enabled, the total number of timeslots supported is 144. The full 6 ports are supported for DS1 and 4.8 ports are supported for E1.

B.1.3.1 Enabling Idle Channel Suppression

Idle channel suppression is enabled on a per-connection basis. The default idle detection mechanism is idle pattern matching. To enable idle channel suppression, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -oport <oport> -ots <ots>
-idlesupp (enable|disable) -oidlesupp (enable|disable)
```

For example:

```
myswitch::configuration ces> new 1a3 1 -oport 1b1 -ots 1 -idlesupp enable
-oidlesupp enable
```

Alternatively, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -oport <oport> -ovpi <ovpi>
-ovci <ovci> -idlesupp (enable|disable) -oidlesupp (enable|disable)
```

For example:

```
myswitch::configuration ces> new 1a3 1 -oport 1b1 -ovpi 0 -ovci 100 -idlesupp
enable -oidlesupp enable
```

B.1.3.2 Configuring Idle Detection

Idle channel suppression is enabled when any idle channel suppression option is used in the AMI command line. An idle detection mechanism is used on a per-connection basis. When idle pattern matching is used, the default pattern is FF. The same default pattern is used for mask pattern matching. When the signalling method is used, the default value is as per Customer Installation to Network idle patterns. The idle detection method is based on the connection type as follows:

- CAS connections use idle CAS signalling pattern matching.
- Structured Basic connections use idle pattern matching.

B.1.3.2.1 Configuring Idle CAS Signalling Pattern Matching

Standard idle signalling patterns are used by names. Only one type can be selected. To configure CAS signalling pattern matching, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>)-oport <oport> -ots <ots>
-oidlesupp (enable|disable) -oidlepat (<idle pattern> | <cas pattern>)
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat fxsluser01
-oport 1b2 -ots 1 -oidlesupp enable -oidlepat em01
```

Alternatively, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>) -oport <oport> -ovpi <ovpi>
-ovci <ovci>
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat em01 -oport 1b2
-ovpi 0 -ovci 100
```

B.1.3.2.2 Configuring an Idle Pattern for Pattern Matching

If an idle pattern are configured, by default, idle channel suppression and idle suppression based on pattern matching is enabled. The idle pattern is used on per-connection basis. A maximum of one pattern can be used for structured basic connections. To configure an idle pattern for pattern matching, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>) -oport <oport> -ots <ots>
-oidlesupp (enable|disable) -oidlepat (<idle pattern> | <cas pattern>)
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat 1F -oport
1b2 -ots 1 -oidlesupp enable -oidlepat 2F
```

Alternatively, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>) -oport <oport> -ovpi <ovpi>
-ovci <ovci>
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat 1F -oport 1b2
-ovpi 0 -ovci 100
```

B.1.3.3 Configuring Mask Patterns

To configure mask patterns, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlemask <idlemask> -idlepat (<idle pattern> | <cas pattern>) -oport <oport>
-ots <ots> -oidlesupp (enable|disable) -oidlemask <idlemask>
-oidlepat (<idle pattern> | <cas pattern>)
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlemask F0 -idlepat
2F -oport 1b2 -ots 1 -oidlesupp enable -oidlemask F0 -oidlepat 2F
```

Alternatively, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlemask <idlemask> -idlepat (<idle pattern> | <cas pattern>) -oport <oport>
-ovpi <ovpi> -ovci <ovci>
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlemask FF -idlepat
7F -oport 1b2 -ovpi 0 -ovci 100
```

B.1.3.4 Configuring an Integration Period

The integration period is provided in μs. Only an active to idle integration period is supported. To configure the integration period, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>) -idleintp <idleintp> -oport <oport>
-ots <ots> -oidlesupp (enable|disable) -oidlepat (<idle pattern> | <cas pattern>)
-oidleintp <idleintp>
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat 2F -idleintp
500 -oport 1b2 -ots 1 -oidlesupp enable -oidlepat 1F -oidleintp 600
```

Alternatively, you could use the following syntax:

```
myswitch::configuration ces> new <port> <timeslots> -idlesupp (enable|disable)
-idlepat (<idle pattern> | <cas pattern>) -idleintp <idleintp> -oport <oport>
-ovpi <ovpi> -ovci <ovci>
```

For example:

```
myswitch::configuration ces> new 1a3 1 -idlesupp enable -idlepat 2F -idleintp
500 -oport 1b2 -ovpi 0 -ovci 100
```

B.2 Configuring CES Connections

The **ces** commands let you create and delete CES connections, as well as display the status of existing connections. You can display the list of available subcommands by typing **?** at the **ces** level.

```
myswitch::configuration ces> ?
  new              delete           show
```

B.2.1 Creating a New CES Connection

To create a new CES connection, you must set several parameters. Enter the following to create a new CES connection:

```
myswitch::configuration ces> new <port> <timeslots>
```

The CES **new** command can also be used as shown below. When the following parameters are used, by default, an appropriate entry is made in the UPC table and a bidirectional PVC is created with the proper UPC index.

```
or: new <port> <timeslots> -oport <oport> -ots <ots>
or: new <port> <timeslots> -oport <oport> -ovpi <ovpi> -ovci <ovci>
```

The following advanced options can be used when creating CES connections:

```
advanced options:
[-srts (on|off)] [-fupc <index>] [-bupc <index>]
[-cas (basic|cas)] [-partialfill <partialfill>] [-reassCDVT <cdvt>]
[-bufSize <bufSize>] [-integ <integ>] [-idlesupp (enable|disable)]
[-idlemask <idlemask>] [-idlepat (<idle pattern> | <cas pattern>)]
[-idleintp <idleintp>] [-oidlesupp (enable|disable)] [-oidlemask <idlemask>]
[-oidlepat (<idle pattern> | <cas pattern>)] [-oidleintp <idleintp>]
```

NOTE

SRTS is only available on unstructured connections, which are created by specifying **`all`** for the *`<timeslots>`* parameter.

The **`-cas`** and **`-partialfill`** options are not applicable to unstructured mode.

Structured mode is selected by indicating the exact timeslots to be used. For example, timeslots 1, 2, and 3 would be entered as 1-3, timeslots 2, 4, and 6 would be entered as 2:4:6, and combinations such as 1-4:9-11:12 are allowed.

The parameters for new are defined as follows:

Parameter	Description
port	The port on which the CES connection is to be created.
timeslots	Indicates which timeslots (1-24 for DS1, 1-31 for E1) are being configured for a particular PVC. **`all`** indicates unstructured service. The time slot assignments may be either contiguous or non-contiguous DS0s.
oport	The output port of the CES connection, which can be a CES port or an ATM port.
ovpi	The output Virtual Path Identifier (VPI) of the CES connection when the output port is not a CES port.
ovci	The output Virtual Channel Identifier (VCI) of the CES connection when the output port is not a CES port.
ots	The output timeslots of the CES connection when the output port is a CES port.
srts	Indicates whether Synchronous Residual Time Stamp (SRTS) clock recovery is to be enabled on this connection. **`on`** indicates that SRTS is enabled, **`off`** indicates that SRTS is disabled. The default is **`off`**.
-fupc <index>	The UPC contract type to be used in the ingress direction of the connection. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about UPC contracts.)
-bupc <index>	The UPC contract type to be used in the egress direction of the connection. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about UPC contracts.)
cas	Indicates whether Channel Associated Signalling (CAS) is to be used on the connection. **`basic`** indicates that CAS will not be used, **`cas`** indicates that CAS will be used. The default is **`basic`**.
partialfill	Indicates how many of the available 47 payload bytes in each cell are used before they are deemed “full” and ready for transmission across the ATM network (i.e., how much of the ATM cell contains data and how much is filler). The range for this parameter is 12 to 47. The default value is **`47`**, for 47 bytes of data. partialfill is used to minimize network transmission latency and is useful especially with time-sensitive, robbed-bit signalling sources.

Parameter	Description
-reassCDVT <cdvt>	The Cell Delay Variation Tolerance for cells being received by the segmentation and reassembly (SAR) engine. The range for this parameter is 100 to 24000 (in µs), and the default is **2000** (i.e., 2 µs).
bufSize	The amount of reassembly buffer space allocated for the connection. The default is **256** bytes per timeslot.
integ	The amount of time allocated to re-establish the connection before, while, or after the call is established, or in the case of interruption. The default is **2500** µs.
-idlesupp (enable | disable)	**enable** means idle channel suppression is going to be used on the incoming side of this connection. **disable** means idle channel suppression is not going to be used on the incoming side of this connection. The default is **disable**. When this option is enabled, a total of 144 timeslots (6 ports for DS1 or 4.8 ports for E1) is supported. When this option is enabled on structured basic connections, a minimum partialfill size of 24 is required.
-idlemask <idlemask>	The mask pattern for idle detection on the incoming side of this connection. This method is only used for structured basic service. The range of values is 01 to FF. The default is **FF**.
-idlepat (<idle pattern> | <cas pattern>)	The pattern for idle detection on the incoming side of this connection. For detection based on both idle and mask patterns, it contains the idle octet pattern. For detection based on signalling, it contains one of the following CAS patterns: em00, em01, fxolsuser01, fxolsuser11, fxolsnet00, fxolsnet01, fxslsuser00, fxslsuser01, fxslsnet01, fxslsnet11, fxsgsuser01, fxsgsnet10, fxsgsnet11, fxogsuser10, fxogsuser11, fxogsnet01, r210. A maximum of one idle pattern can be used for structured basic connections. Idle patterns are filled from the least significant byte. The default is **FF** for structured basic connections. The default CAS idle AB bit pattern is **em00**.
-idleintp <idleintp>	The integration period for idle detection on the incoming side of this connection. Idle patterns are observed for this period before declaring that an active connection has gone idle. For both CAS and basic connections, the range of values is 500 µs to 2 seconds and the default is **1** second.
-oidlesupp (enable | disable)	**enable** means idle channel suppression is going to be used on the outgoing side of this connection. **disable** means idle channel suppression is not going to be used on the outgoing side of this connection. The default is **disable**. When this option is enabled, a total of 144 timeslots (6 ports for DS1 or 4.8 ports for E1) is supported. When this option is enabled on structured basic connections, a minimum partialfill size of 24 is required.
-oidlemask <idlemask>	The mask pattern for idle detection on the outgoing side of this connection. This method is only used for structured basic service. The range of values is 01 to FF. The default is **FF**.
-oidlepat (<idle pattern> | <cas pattern>)	The patterns for idle detection on the outgoing side of this connection. For detection based on both idle and mask patterns, it contains the idle octet patterns. For detection based on signalling, it contains one of the following CAS patterns: em00, em01, fxolsuser01, fxolsuser11, fxolsnet00, fxolsnet01, fxslsuser00, fxslsuser01, fxslsnet01, fxslsnet11, fxsgsuser01, fxsgsnet10, fxsgsnet11, fxogsuser10, fxogsuser11, fxogsnet01, r210. A maximum of one idle pattern can be used for structured basic connections. Idle patterns are filled from the least significant byte. The default is **FF** for structured basic connections. The default CAS idle AB bit pattern is **em00**.

Parameter	Description
-oidleintp <idleintp>	The integration period for idle detection on the outgoing side of this connection. Idle patterns are observed for this period before declaring that an active connection has gone idle. For both CAS and basic connections, the range of values is 500 µs to 2 seconds and the default is **1** second.

B.2.2 Displaying CES Connections

To display the current CES connections, enter the following:

```
myswitch::configuration ces> show
  CES           Input                                  Output
  Service State Port  Timeslots    VPI VCI  Type Port TimeSlots    VPI VCI
  24      down  1A1   1            0   129  -    -    -            -   -
  31      down  1A1   2-3          0   130  spvc 1D3  -            0   32
  32      down  1A1   4-5          0   131  pvc  1D4  -            0   150
  33      down  1A1   6-7          0   132  spvc 1D3  -            0   35
```

The fields in this display are defined as follows:

Field	Description
CES Service	The identification number (assigned by the switch) of this CES connection.
State	Indicates whether the CES connection is enabled (`up`) or disabled (`down`).
Input Port	The incoming port on which the CES connection exists.
Timeslots	Indicates which timeslots (1-24 for DS1, 1-31 for E1) are configured for the input port. **`all`** indicates unstructured service.
Input VPI	The incoming VPI value of the connection.
Input VCI	The incoming VCI value of the connection.
Type	The type of ATM connection (i.e., PVC or SPVC) that is associated with the CES connection.
Output Port	The outgoing port on which the CES connection exists.
Timeslots	Indicates which timeslots (1-24 for DS1, 1-31 for E1) are configured for the output port. **`all`** indicates unstructured service.
Output VPI	The outgoing VPI value of the connection.
Output VCI	The outgoing VCI value of the connection.

To display the advanced settings of the current CES connections, enter the following:

```
myswitch::configuration ces> show advanced
                     Service     Clock       Partial     Max        Integ.
Service MapVPI MapVCI Type       Mode  Cas   Fill    BufSize   CDVT Period
2024    0      129    structured synch basic 0           256    900   2500
```

The fields in this display are defined as follows:

Field	Description
CES Service	The identification number (assigned by the switch) of this CES connection.
MapVPI	The incoming VPI value of the connection.
MapVCI	The incoming VCI value of the connection.
Service Type	Shows if this connection uses structured or unstructured service.
Clock Mode	**Synch** means that the connection is in synchronous mode (either structured or unstructured). **SRTS** means that the connection is in asynchronous (unstructured) mode. (Synchronous Residual Time Stamp (SRTS) clock recovery is enabled on this connection.)
Cas	**basic** indicates that Channel Associated Signalling (CAS) will not be used, **cas** indicates that CAS will be used.
Partial Fill	Indicates how many of the available 47 payload bytes in each cell are used before they are deemed “full” and ready for transmission across the ATM network (i.e., how much of the ATM cell contains data and how much is filler). partialfill is used to minimize network transmission latency and is useful especially with time-sensitive, robbed-bit signalling sources.
Max BufSize	The amount of reassembly buffer space allocated for the connection. The default is 256 bytes per timeslot.
CDVT	The Cell Delay Variation Tolerance for cells being received by the segmentation and reassembly (SAR) engine. The range for this parameter is 100 to 24000 (in μs), and the default is 2000 (i.e., 2 μs).
Integ. Period	The amount of time allocated to re-establish the connection before, while, or after the call is established, or in the case of interruption. The default is 2500μs.

To display the idle channel suppression information for the current CES connections, enter the following:

```
myswitch::configuration ces> show [<id>] [(advanced|idlesupp)]
myswitch::configuration ces> show idlesupp

CES     Idlesupp   Idle    Idle  NoOf  Idle      IdleInt.
Service State      Type    Mask  Patt  Patterns  Period(µs)
20024   disabled    -       -      -   -              -
20031   enabled   pattern  FF      1   7F           500
20032   enabled    cas     01b     1   10b          100
20033   disabled    -       -      -   -              -
20034   disabled    -       -      -   -              -
20035   enabled   pattern  FF      1   7F           500
20036   enabled    cas     11      1   00           100
20037   disabled    -       -      -   -              -
```

The fields in this display are defined as follows:

Field	Description
CES Service	The identification number (assigned by the switch) of this CES connection.
Idlesupp State	**enabled** means idle channel suppression is going to be used on this connection. **disabled** means idle channel suppression is not going to be used on this connection.
Idle Type	The idle detection mechanism used on this connection. **pattern** means idle detection is based on idle patterns. **cas** means idle detection is based on CAS signalling. This field is read-only.
Idle Mask	The mask pattern for idle detection on this connection. This method of idle detection can only be used on basic and unstructured connections.
NoOf Patt	The number of idle patterns configured for detection. This field is read-only.
Idle Patterns	The patterns for idle detection on this connection. For detection based on both idle and mask patterns, it contains the idle octet patterns. For detection based on signalling, it contains one of the following CAS patterns: em00, em01, fxolsuser01, fxolsuser11, fxolsnet00, fxolsnet01, fxslsuser00, fxslsuser01, fxslsnet01, fxslsnet11, fxsgsuser01, fxsgsnet10, fxsgsnet11, fxogsuser10, fxogsuser11, fxogsnet01, r210. A maximum of one idle pattern can be used for structured basic connections. Idle patterns are filled from the least significant byte.
IdleInt. Period (µs)	The integration period for idle detection on this connection. Idle patterns are observed for this period before declaring that an active connection has gone idle.

APPENDIX C Converting FT-PNNI and PNNI Networks

This appendix discusses the conversion of both non-hierarchical and hierarchical FT-PNNI networks to ATM Forum PNNI (hereafter referred to as PNNI) routing. It also describes the conversion of a non-hierarchical PNNI network to a lop-sided hierarchical PNNI network. It is assumed that you are familiar with the fundamentals of FT-PNNI and PNNI routing and familiar with FORE's implementation of PNNI as described in Chapter 6 of this manual.

The first section discusses PNNI routing in networks that contain *ASX*-1000, *ASX*-1200, or *TNX*-1100 switches. The various procedures for converting your network are described in the later sections. If you do not have any *ASX*-1000, *ASX*-1200, or *TNX*-1100 switches in your network, you can skip to Section C.2 or Section C.3 to learn how to convert your network from FT-PNNI to PNNI. You can skip to Section C.4 to learn how to convert your non-hierarchical PNNI network to a lop-sided hierarchical PNNI network. If you do have *ASX*-1000, *ASX*-1200, or *TNX*-1100 switches in your network, it is recommended that you read Section C.1 first to avoid potential configuration problems when migrating your network.

- **Section C.1 -** ASX-1000, ASX-1200, and TNX-1100 Routing Configuration Issues
- **Section C.2 -** Migration of a Non-Hierarchical FT-PNNI Network
- **Section C.3 -** Migration of a Hierarchical FT-PNNI Network
- **Section C.4 -** Migration of a Non-hierarchical PNNI Network to a Lop-sided Hierarchical PNNI Network

C.1 *ASX*-1000, *ASX*-1200, and *TNX*-1100 Routing Configuration Issues

This section discusses routing in hierarchical networks in which there are *ASX*-1000, *ASX*-1200, or *TNX*-1100 switches present in your network. Because each one of the four fabrics appears as a single node in routing and because you cannot take a two-hop path to go from one switch fabric to another switch fabric in the same *ASX*-1000, *ASX*-1200, or *TNX*-1100, there are certain FT-PNNI and PNNI node configurations that should be avoided. These special configurations are addressed individually and alternate configurations are suggested.

C.1.1 *ASX*-1000s, *ASX*-1200s, or *TNX*-1100s in FT-PNNI Peer Groups

Figure C.1 shows an invalid configuration of an *ASX*-1000, *ASX*-1200, or *TNX*-1100 divided between two FT-PNNI peer groups called A and B. The *ASX*-1000's, *ASX*-1200's, or *TNX*-1100's four FT-PNNI nodes are A.1 and A.2 in peer group A and B.1 and B.2 in peer group B. From peer group A, the intra-peer group links of peer group B are not visible. So, the link B.1 to B.2 is not visible in peer group A. Similarly, non-border nodes B.3 and B.4 belonging to peer group B are not visible in peer group A.

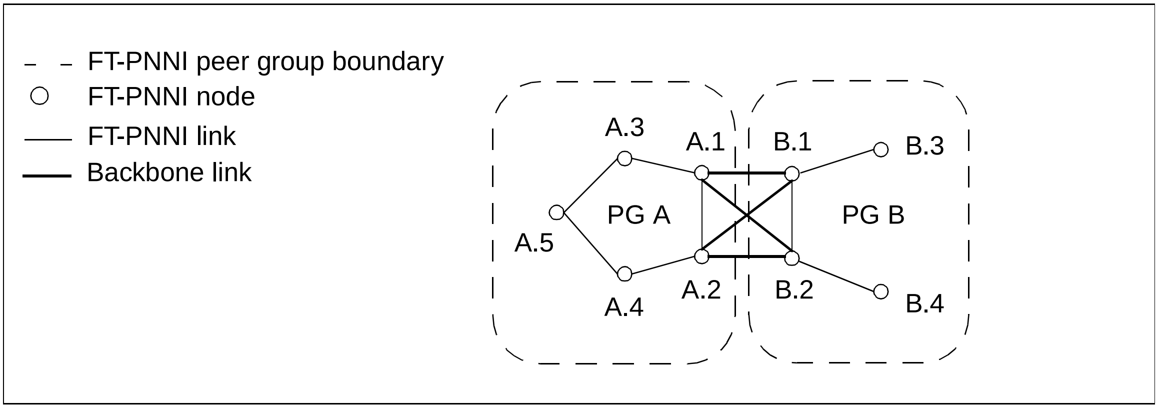

Figure C.1 - Invalid Configuration of *ASX*-1000, *ASX*-1200, or *TNX*-1100 Split between Two FT-PNNI Peer Groups

Suppose you want to route from A.4 to B.3. After the initial step of prefix matching, the Peer Group Summary Node (PGSN) of peer group B (not shown in the figure) is chosen as the destination for this path computation. Since B.1 and B.2 both advertise links to the PGSN and they are equidistant from A.4, A.4 may decide to construct the Destination Transit List (DTL) as A.4 to A.2 to B.2 to B's PGSN. When the setup of this call proceeds into peer group B and reaches node B.2, the only way to route to B.3 is through B.1. But since you cannot take a two-hop path across the *ASX*-1000, *ASX*-1200, or *TNX*-1100, this call setup will fail.

This problem can be avoided by not breaking up an *ASX*-1000, *ASX*-1200, or *TNX*-1100 between multiple FT-PNNI peer groups. For example, in the network in Figure C.2, the entire *ASX*-1000, *ASX*-1200, or *TNX*-1100 could have been made part of peer group B with only A.3, A.4, and A.5 in peer group A. A.3 and A.4 would be the border nodes in peer group A connected to A.1 and A.2 (re-numbered to have B as their peer group ID) in peer group B.

C.1.2 *ASX*-1000s, *ASX*-1200s, or *TNX*-1100s in PNNI Areas

The problem described in Section C.1.1 can also occur in a network of two PNNI peer groups (areas). For example, in the poorly-constructed network shown in Figure C.2, the fabric on the top left (fabric 1) is configured as a split switch with two nodes, one in peer group A and the other in peer group B. All of the other fabrics have one node each. The node on the bottom left fabric (fabric 2) is in peer group A and the nodes in the other two fabrics (3 and 4) are in peer group B. The links between the fabric 2 and the fabrics 3 and 4 do not come up because the peer group IDs of the PNNI nodes connected these links do not match.

NOTE This example assumes that there is no PNNI Logical Group Node (LGN) hierarchy configured in this network.

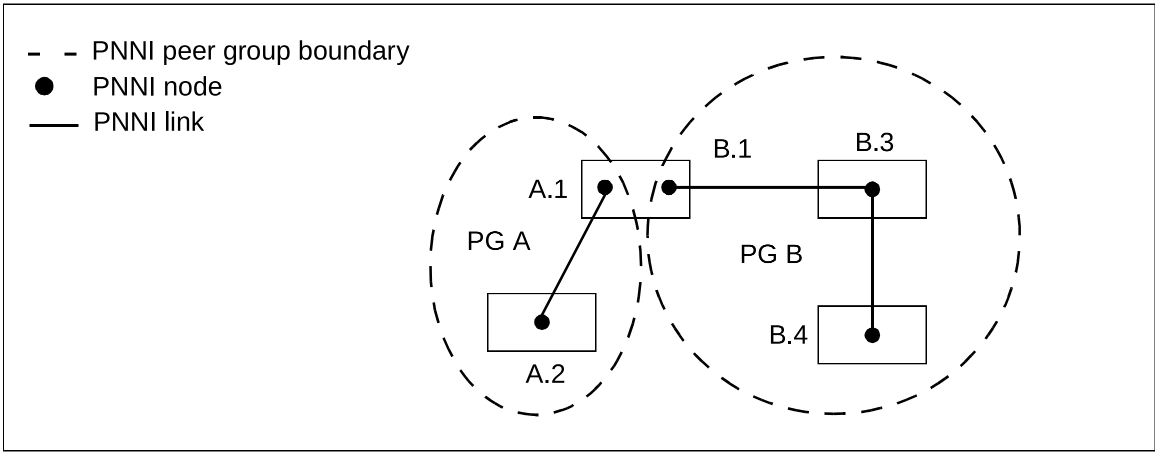

Figure C.2 - Invalid Configuration of *ASX*-1000, *ASX*-1200, or *TNX*-1100 Split between Two PNNI Peer Groups

The problem in the network shown above occurs when computing a path from A.2 to B.3. In this case, routing is forced to use a two-hop path across the backplane, thereby failing to set up a call. To avoid this problem, be sure that you configure all fabrics in an *ASX*-1000, *ASX*-1200, or *TNX*-1100 as part of the same peer group.

C.1.3 Multiple Gateways in an *ASX*-1000, or *TNX*-1100

Another potential problem can occur when an *ASX*-1000, *ASX*-1200, or *TNX*-1100 in a FT-PNNI area has gateways (split-switches with one FT-PNNI and one PNNI node) to a PNNI area. The *ASX*-1000, *ASX*-1200, or *TNX*-1100 can only have, at most, one of its fabrics as a gateway between the FT-PNNI area and the PNNI area. This is because if two or more of the fabrics have gateways to the PNNI area, then the link connecting any two of the gateways will come up as a PNNI link in the PNNI area. This will divide up the backplane links between two different peer groups. This invalid configuration is shown in Figure C.3.

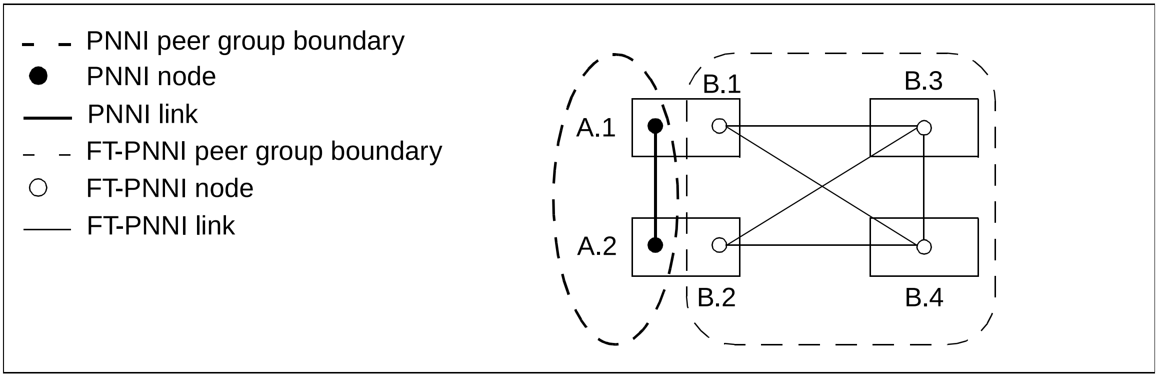

Figure C.3 - Multiple Fabrics of an *ASX*-1000, *ASX*-1200, or *TNX*-1100 Incorrectly Configured as Gateways

In the above figure, the *ASX*-1000, *ASX*-1200, or *TNX*-1100 shown is predominantly part of the FT-PNNI peer group B. However, the *ASX*-1000, *ASX*-1200, or *TNX*-1100 has gateways configured to the PNNI peer group A. Since fabric 1 and fabric 2 are both configured as gateways, the link between them comes up as a PNNI link in peer group A. Because of this, the backplane links of the *ASX*-1000, *ASX*-1200, or *TNX*-1100 are divided between peer group A and peer group B. This may cause two-hop paths to be taken across the backplane links and is an invalid configuration. To avoid this problem, be sure that you configure either only one of the fabrics on an *ASX*-1000, *ASX*-1200, or *TNX*-1100 as a gateway between a single FT-PNNI area and a single PNNI area, or all of the fabrics on an *ASX*-1000, *ASX*-1200, or *TNX*-1100 as a gateway between the same FT-PNNI area and the same PNNI area.

C.1.4 Migrating from FT-PNNI to PNNI Routing

The migration of FT-PNNI networks to PNNI networks is discussed in the following sections. It is possible that while migrating a network containing *ASX*-1000, *ASX*-1200, or *TNX*-1100 switches, invalid configurations similar to the ones described in this section may occur during the intermediate steps. To avoid these potential problems, migrate all of the fabrics in an *ASX*-1000, *ASX*-1200, or *TNX*-1100 at once, keeping them in the same peer groups or areas.

C.2 Migration of a Non-Hierarchical FT-PNNI Network

This section discusses the conversion of a non-hierarchical FT-PNNI network to a non-hierarchical (single peer group) PNNI network. It is assumed that all switches in your network are FORE switches.

C.2.1 Migration Overview

The following basic steps are involved in converting your network. Each of these steps is described in detail in the following section. It is recommended that you read the entire section before attempting to change over your network.

1. Upgrade each switch in the network to *ForeThought* 6.0.x.
2. Choose a switch that is at the edge of the network. Convert it to a gateway switch by changing the default protocol of this switch's default domain to `gateway` and reboot the switch. Upon rebooting, this switch will come up with an FT-PNNI node and a PNNI node. The PNNI node is isolated at this point and does not have any links attached to it.
3. Choose a switch adjacent to the gateway switch, convert it to a gateway switch by changing its default protocol for the default domain to `gateway`, and reboot this switch. It will come up with a FT-PNNI node and a PNNI node. The link between the two gateway switches is attached on either end to the respective PNNI nodes. This link is now the first PNNI link in the network. This single link constitutes a PNNI area connected by the two gateway switches to the FT-PNNI area. The FT-PNNI area now contains all the links in the network except the one link between the two gateways. (It is assumed that there is only one link between the two gateway switches. If there are multiple links, then all of them will become PNNI links after this step).
4. Choose another switch that is adjacent to either one of the two switches already converted to gateway and repeat step step 3. As more and more switches are converted to be gateways, the PNNI area becomes progressively larger and the FT-PNNI area becomes smaller.
5. For each gateway, when the last link gets converted from FT-PNNI to PNNI (i.e., when the last FT-PNNI switch directly connected to this switch becomes PNNI), modify the default protocol of the default domain to PNNI and reboot the switch. By doing this, the FT-PNNI node on this gateway, which no longer has any link, will be deactivated.

C.2.2 Detailed Migration Example

This section gives a detailed example of how to change the example non-hierarchical FT-PNNI network shown in Figure C.4 to a PNNI network.

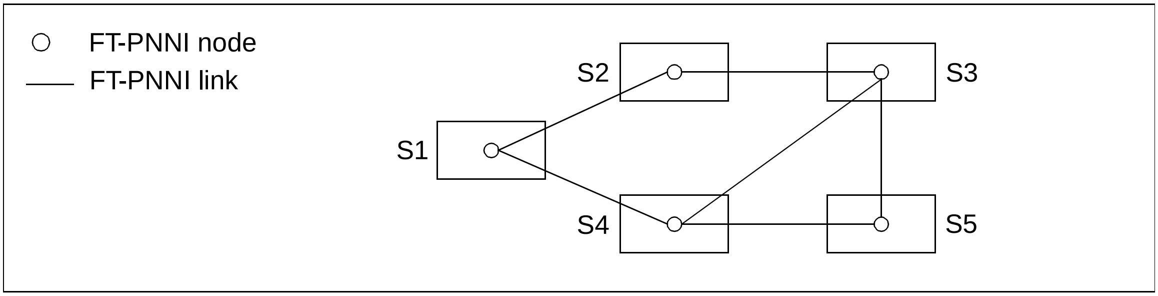

Figure C.4 - A Non-Hierarchical FT-PNNI Network

This network has five switches named S1 through S5. The following steps are involved in changing over this network.

1. Upgrade all of the switches to *ForeThought* 6.0.x using the following AMI command:

   ```
   oper upgrade <remotehost>:<full path to remotefile>
   ```

   For more information about upgrading your switches, see Chapter 4 of the *ATM Switch Installation and Maintenance Manual.*

2. Convert S5 to a gateway switch by modifying the default protocol of the default domain in S5 to `gateway` using the following AMI command:

   ```
     conf atmroute domain modify <domain ID> gateway
   ```

   Reboot switch S5. Upon the reboot, S5 will come up with two nodes: one FT-PNNI node and one PNNI node. By default, these nodes are in areas 4 and 5, respectively, and levels 4 and 5, respectively. At this point, the nodes in the network are divided between the two areas with the PNNI node being the only node in area 5 as shown in Figure C.5.

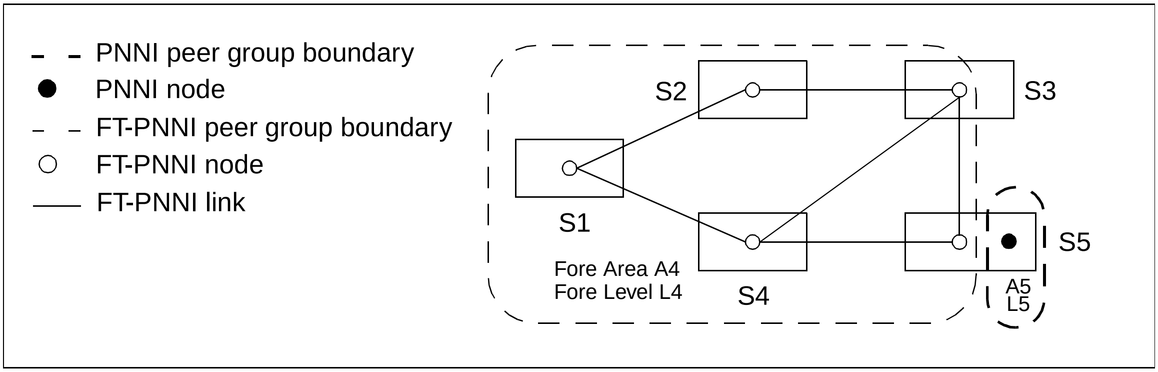

Figure C.5 - S5 Changed to a Gateway Switch

3. Convert S3 to a gateway switch by modifying the default protocol of the default domain in S3 to `gateway` using the following AMI command:

```
conf atmroute domain modify <domain ID> gateway
```

Reboot switch S3. Upon the reboot, S3 will come up with two nodes: one FT-PNNI node in area 4 and one PNNI node in area 5. The link between the two gateway switches S5 and S3 will come up as a PNNI link as shown in Figure C.6. This is the first PNNI link in the network. All of the other links are still FT-PNNI links.

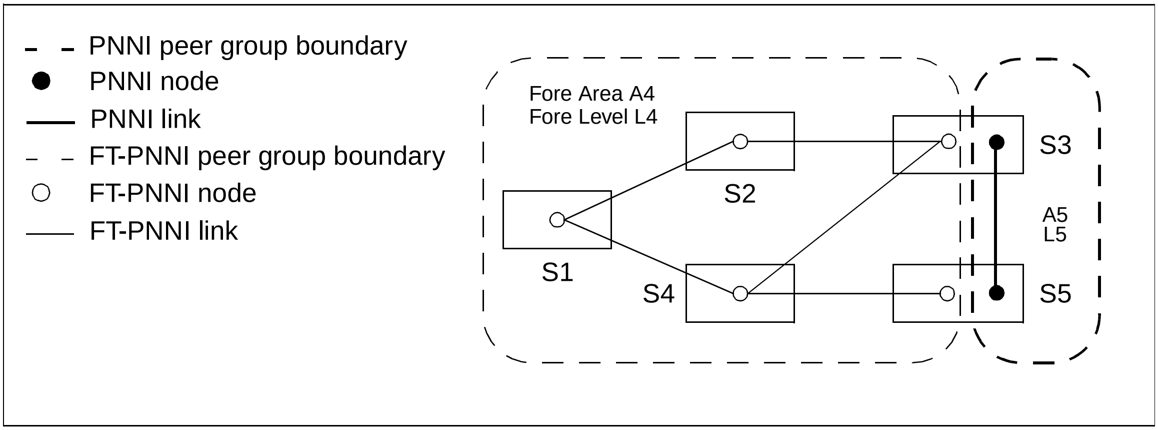

Figure C.6 - S3 Changed to a Gateway Switch

4. Convert S4 to a gateway switch by modifying the default protocol of the default domain in S4 to `gateway` using the following AMI command:

   conf atmroute domain modify *<domain ID>* **gateway**

   Reboot switch S4. Upon the reboot, S4 will come up with a FT-PNNI node in area 4 and a PNNI node in area 5. The link between S4 and S5 will come up as a PNNI link, and the link between S3 and S4 will come up as a PNNI link, since they are both gateway switches.

5. At this point, S5 no longer has any links attached to its FT-PNNI node, so it does not need to be a gateway switch anymore. Modify the default protocol of S5 to `pnni` using the following AMI command:

   conf atmroute domain modify *<domain ID>* **pnni**

   Reboot switch S5. The state of the network at this point is shown in Figure C.7.

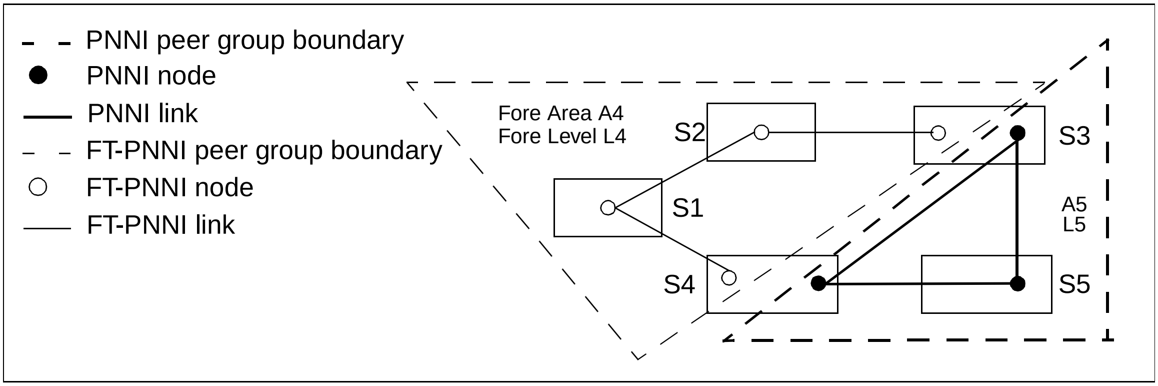

Figure C.7 - S4 Changed to a Gateway Switch and S5 to PNNI

6. Convert S2 to a gateway switch by modifying the default protocol of the default domain in S2 to `gateway` using the following AMI command:

   conf atmroute domain modify *<domain ID>* **gateway**

   Reboot switch S2. Upon the reboot, S2 will come up with a FT-PNNI node in area 4 and a PNNI node in area 5. The only FT-PNNI links left in the network are between S1 and S2 and between S1 and S4.

7. At this point, S3 no longer has any links attached to its FT-PNNI node, so it does not need to be a gateway switch anymore. Modify the default protocol of S3 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch S3.

8. Since S1 is the last switch in the network to have its default protocol modified, it can be changed directly to a PNNI switch rather than to a gateway switch first and then to a PNNI switch. So, modify the default protocol of S1 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch S1. Now all of the links have been switched over to PNNI.

9. Convert both S2 and S4 to PNNI by modifying the default protocol of both S2 and S4 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switches S2 and S4. The final state of the network upon completion of the conversion is shown in Figure C.8.

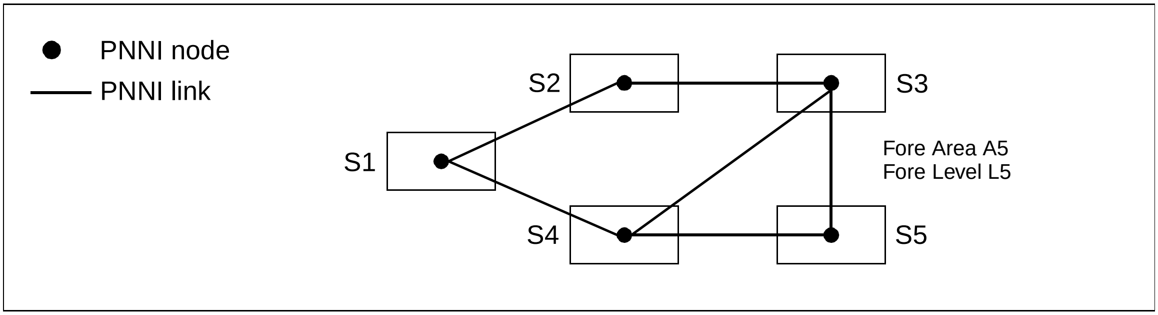

Figure C.8 - A Completely Converted PNNI Network

C.3 Migration of a Hierarchical FT-PNNI Network

This section discusses the migration of a hierarchical FT-PNNI network to a FORE hierarchical PNNI network. It is assumed that all switches in your network are FORE switches. If your network contains a contiguous backbone, use the instructions found in Section C.3.1. If your network does not contain a contiguous backbone, use the instructions found in Section C.3.2.

C.3.1 Migration of a Hierarchical FT-PNNI Network with a Contiguous Backbone

There are two different ways to change a FT-PNNI network with a contiguous backbone. If you want to convert your network starting with the backbone, use the method described in Section C.3.1.1. If you want to convert your network starting with the peer groups, use the method described in Section C.3.1.2. You can use either method.

C.3.1.1 Migration Starting with the Backbone

This section gives a detailed example of how to convert a hierarchical FT-PNNI network with a contiguous backbone as shown in Figure C.9 to a PNNI network. In this figure, only the border nodes are shown. The individual nodes within each peer group are not shown.

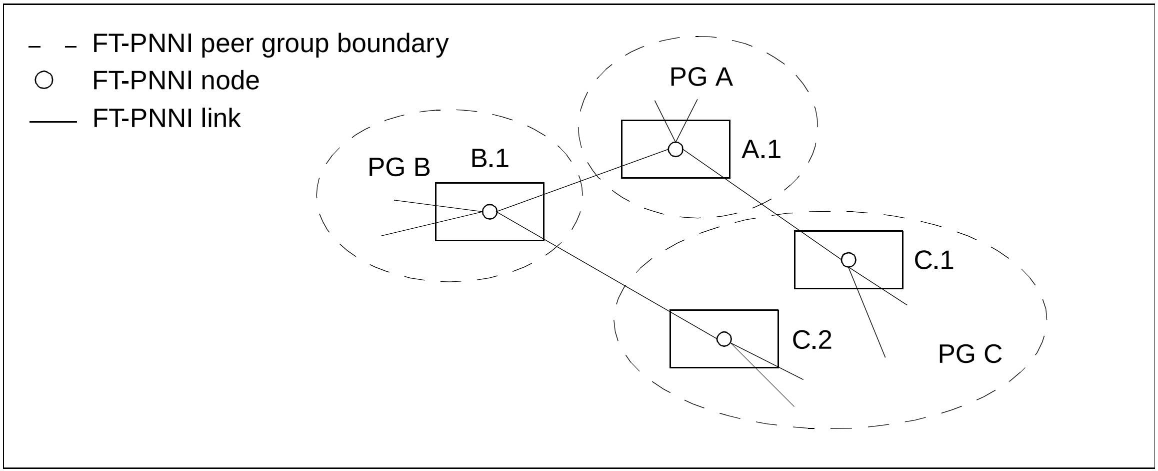

Figure C.9 - Hierarchical FT-PNNI Network with 3 Peer Groups and a Contiguous Backbone

C.3.1.1.1 Migration Overview

The following basic steps are involved in the migration. Each of these steps is described in detail in the following sections. It is recommended that you read the entire section before attempting to change over your network.

1. Upgrade each switch in the network to *ForeThought* 6.0x.
2. Convert the backbone of the FT-PNNI network.
3. Convert the individual peer groups of the FT-PNNI network.

C.3.1.1.1.1 Upgrade the Switches

Upgrade all of the switches to *ForeThought* 6.0.x using the following AMI command:

```
oper upgrade <remotehost>:<full path to remotefile>
```

For more information about upgrading your switches, see Chapter 4 of the *ATM Switch Installation and Maintenance Manual*.

After the upgrade, there is only one FT-PNNI area with area ID 4 and level 4. The three FT-PNNI peer groups are contained within this one area.

C.3.1.1.1.2 Convert the Backbone

C.1 will be the first switch to be converted to a gateway between the existing FT-PNNI area and the PNNI area to be created. Since the backbone is being migrated to PNNI first, the backbone PNNI area should have a higher level (numerically smaller) than the FT-PNNI area. The level of the backbone will be 2.

1. Change the peer group ID of the PNNI node on C.1 to D (so that it has a new peer group ID which is different from the peer group IDs of the three existing FT-PNNI peer groups: A, B, and C). Although the PNNI node in C.1 will not be activated until the default protocol gets changed to gateway, it is possible to change the level of the PNNI node while it is still down. So, change the level of the PNNI node at switch C.1 to 2. Use the following AMI command to change both:

```
conf atmroute pnni node modify <index> -pgid d -forelevel 2
```

2. Modify the default protocol of the default domain in C.1 to `gateway` using the following AMI command:

```
conf atmroute domain modify <domain ID> gateway
```

   Reboot switch C.1. Upon the reboot, C.1 will come up with two nodes: one FT-PNNI node and one PNNI node. There are now two areas: one FT-PNNI area 4 at level 4, and one PNNI area 5 at level 2.

3. A.1 will be the second switch to be converted to a gateway between the existing FT-PNNI area and the PNNI area to be created. Change the peer group ID of its PNNI node (currently down) to D, modify the level of its PNNI node to 2, and modify the default protocol of the default domain in A.1 to `gateway` using the following AMI commands:

```
conf atmroute pnni node modify <index> -pgid d -forelevel 2

      conf atmroute domain modify <domain ID> gateway
```

Reboot switch A.1. Upon the reboot, the link between the PNNI nodes on C.1 and A.1 now becomes the first PNNI link in the network. The state of the network at this stage is shown in Figure C.10.

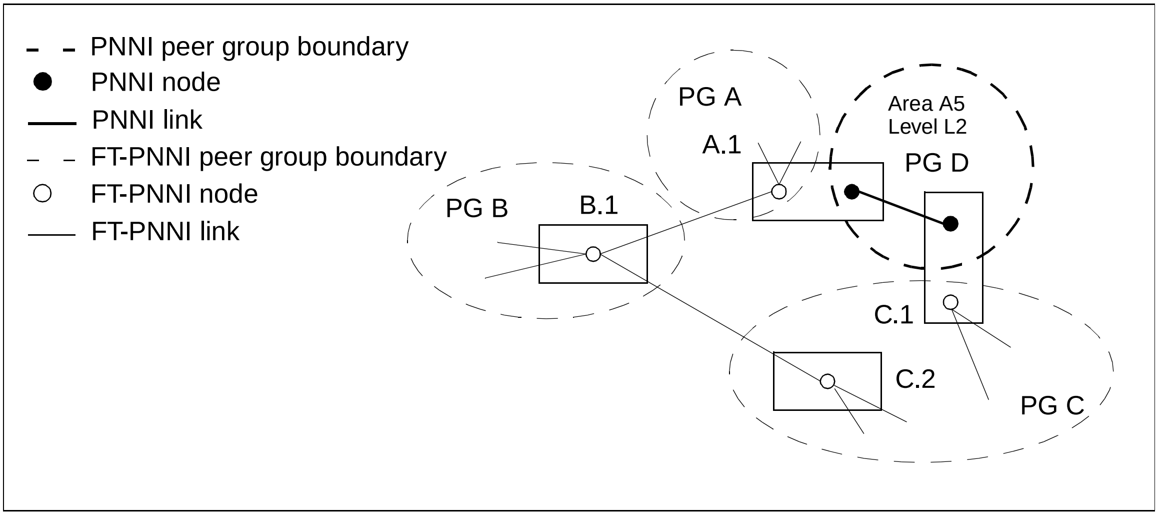

Figure C.10 - C.1 and A.1 as Gateway Switches

4. B.1 will be the next switch to be converted to a gateway switch. Administer the node down. Change the peer group ID of its PNNI node to D, modify the level of its PNNI node to 2, modify the default protocol of the default domain in B.1 to `gateway`, then administer the node up using the following AMI commands:

```
conf atmroute pnni node admin <index> down
```

```
conf atmroute pnni node modify <index> -pgid d -forelevel 2
```

```
conf atmroute domain modify <domain ID> gateway
```

```
conf atmroute pnni node admin <index> up
```

   Reboot switch B.1. Upon the reboot, the link between the A.1 and B.1 now becomes a PNNI link.

5. At this point, the FT-PNNI peer group A is now severed from the rest of the FT-PNNI area. Since there are now two areas with area ID 4, one of them has to be renamed. Change the area ID of the FT-PNNI node on A.1 to 2 using the following AMI command:

```
conf atmroute ftpnni forearea 2
```

   Reboot A.1 so the change takes effect. Figure C.11 shows the state of the network at this point.

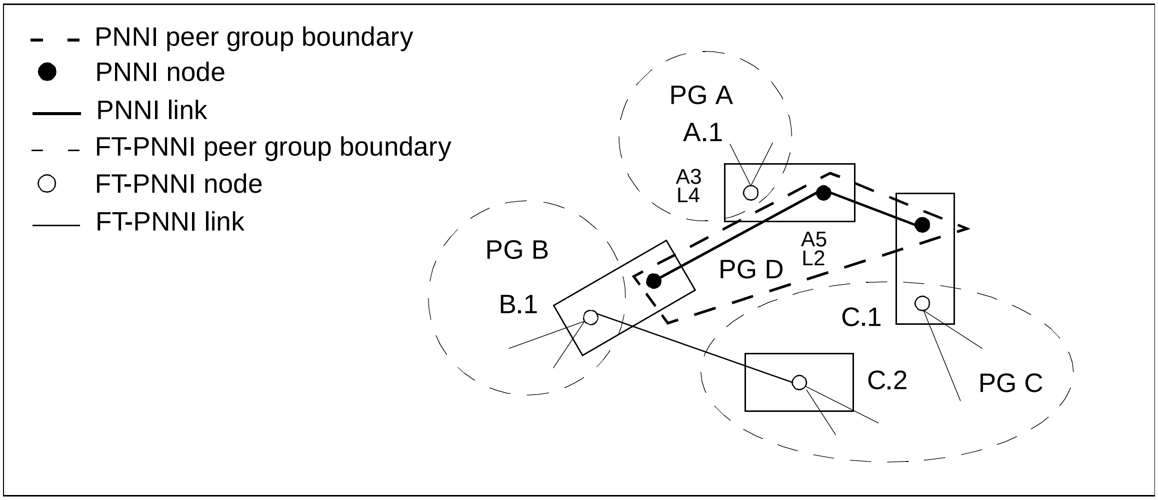

Figure C.11 - Peer Group Severed from the Rest of the FT-PNNI Area

6. C.2 is the final backbone switch to be converted to a gateway switch. Change the peer group ID of its PNNI node (currently down) to D, modify the level of its PNNI node to 2, and modify the default protocol of the default domain in C.2 to `gateway` using the following AMI commands:

```
conf atmroute pnni node modify <index> -pgid d -forelevel 2
      conf atmroute domain modify <domain ID> gateway
```

   Reboot switch C.2. Upon the reboot, all of the backbone links of the network are converted to PNNI.

7. At this point, the FT-PNNI peer group B is an area by itself. So, the area ID of the FT-PNNI node on B.1 needs to be changed to 3 using the following AMI command:

```
conf atmroute ftpnni forearea 3
```

   Reboot switch B.1. The FT-PNNI peer group C is also an area by itself, but it can be allowed to retain its original area ID of 4. Upon the reboot of switch B.1, the migration of the backbone is complete.

C.3.1.1.1.3 Convert the Individual Peer Groups

Now each individual peer group needs to be converted to PNNI. In this case, peer group C is being used as an example. The other peer groups would be converted using the same steps outlined here. Figure C.12 shows an example configuration of the FT-PNNI nodes within peer group C that need to be converted.

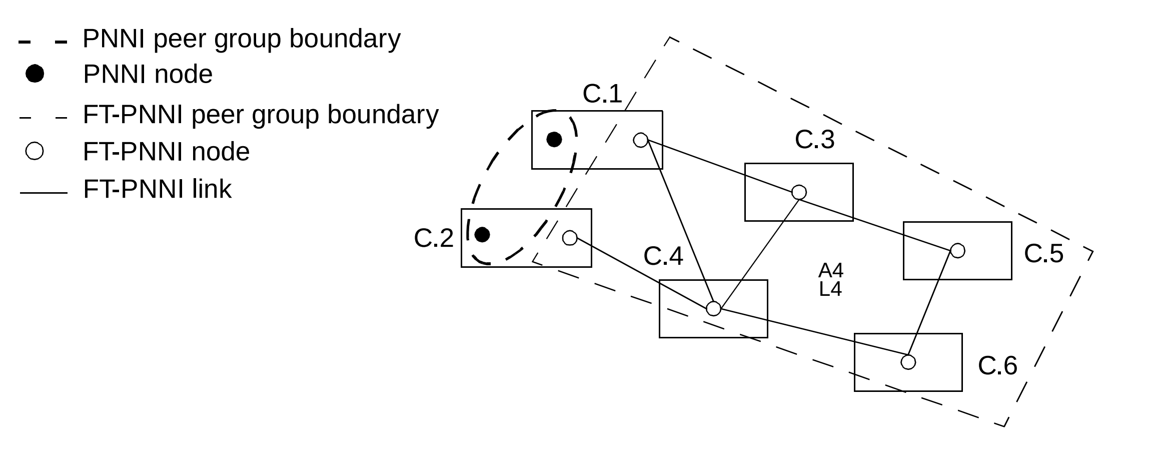

Figure C.12 - Peer Group C before the Conversion to PNNI

C.6 will be the first switch to be converted to a gateway between the existing FT-PNNI area and the PNNI area. This area should be at a lower level. The level of this peer group will be 6.

1. Modify the level of the PNNI node (currently down) in C.6 to 2, change the area ID of its PNNI node to 6, and modify the default protocol of the default domain in C.6 to `gateway` using the following AMI commands:

   ```
   conf atmroute pnni node modify <index> -forelevel 6 -forearea 6
   
           conf atmroute domain modify <domain ID> gateway
   ```

   Reboot switch C.6. Upon the reboot, the first PNNI node is created in area 6. At this point, this node has no links attached to it.

2. Convert C.5 to a gateway switch by changing the level of its PNNI node to 6, changing the area ID of C.5 to 6, and modifying the default protocol of the default domain in C.5 to `gateway` using the following AMI commands:

   ```
   conf atmroute pnni node modify <index> -forelevel 6 -forearea 6
   
           conf atmroute domain modify <domain ID> gateway
   ```

   Reboot switch C.6. Upon the reboot, the first PNNI link in area 6 comes up between switches C.6 and C.5.

3. Convert C.4 to a gateway switch by changing the level of its PNNI node to 6, changing the area ID of C.4 to 6, and modifying the default protocol of the default domain in C.4 to `gateway` using the following AMI commands:

   ```
   conf atmroute pnni node modify <index> -forelevel 6 -forearea 6
   
           conf atmroute domain modify <domain ID> gateway
   ```

   Reboot switch C.4. Upon the reboot, the link between C.4 and C.6 comes up as PNNI in area 6. The links between C.4 and C.2 and between C.4 and C.1 will attempt to come up as PNNI (because these links are between two gateway switches), but they will not become operational (because the PNNI links will not reach the two-way-inside hello state).

4. Since C.6 does not need to be a gateway switch anymore, change the default protocol of the default domain in C.6 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.6.

5. Since C.3 is the last FT-PNNI switch, convert it directly to a PNNI switch by changing the level of its PNNI node to 6, changing the area ID of C.3 to 6, and modifying the default protocol of the default domain in C.3 to `pnni` using the following AMI commands:

```
conf atmroute pnni node modify <index> -forelevel 6 -forearea 6

      conf atmroute domain modify <domain ID> pnni
```

Reboot switch C.3. Upon the reboot, the link between C.3 and C.1 will attempt to become PNNI, but will not become operational. At this point, the peer group is temporarily separated from the backbone and other peer groups in the network as shown in Figure C.13. It will be restored to full connectivity when C.1 and C.2 have their second PNNI nodes created.

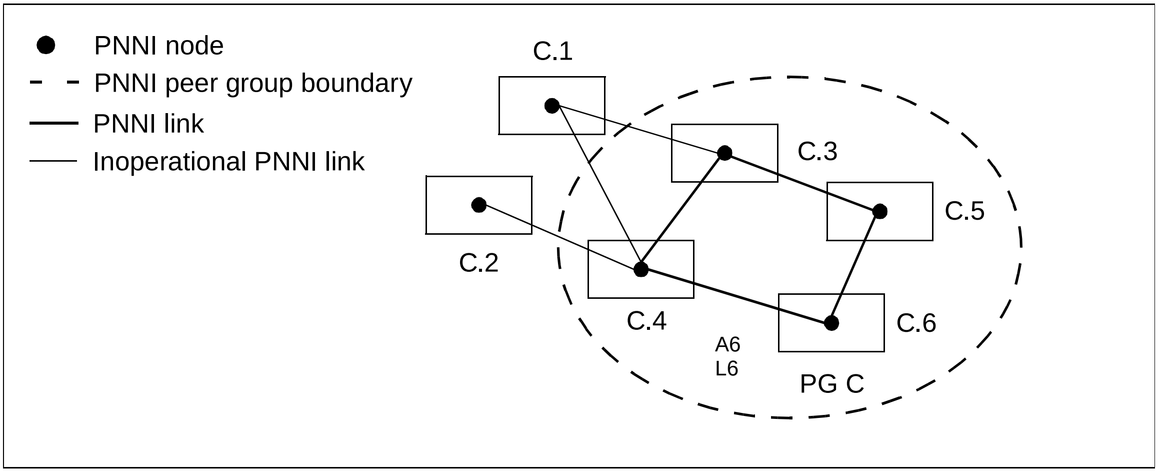

Figure C.13 - C.1 and C.2 Not Part of Peer Group C

6. Since C.5 and C.4 do not need to be gateway switches anymore, change the default protocol of the default domain in C.5 and in C.4 to `pnni` using the following AMI command:

```
conf atmroute domain modify <domain ID> pnni
```

Reboot switch C.5 and switch C.4.

7. Change the default protocol of the default domain in C.1 to `pnni` using the following AMI command:

    ```
    conf atmroute domain modify <domain ID> pnni
    ```

    Reboot switch C.1.

8. Create a second PNNI node in C.1 with a node index of 2, the peer group ID set to C, area 6, and level 6 using the following AMI commands:

```
conf atmroute pnni node modify 2 -pgid c -forelevel 6 -forearea 6
```

```
conf atmroute domain modify <domain ID> pnni
```

Reboot switch C.1. Then, modify the interfaces on C.1 corresponding to the links to C.4 and C.3 and attach them to node 2 on C.1. Use the following AMI command for each link:

```
conf atmroute pnni interface modify <port> <vpi> -nodeix 2
```

This brings the links to C.3 and C.4 back to being operational.

9. Change the default protocol of the default domain in C.2 to `pnni` using the following AMI command:

    ```
    conf atmroute domain modify <domain ID> pnni
    ```

    Reboot switch C.2.

10. Create a second PNNI node in C.2 with a node index of 2, the peer group ID set to C, area 6, and level 6 using the following AMI commands:

```
conf atmroute pnni node modify 2 -pgid c -forelevel 6 -forearea 6
```

```
conf atmroute domain modify <domain ID> pnni
```

Reboot switch C.2.

11. Modify the interfaces on C.2 corresponding to the link to C.4 and attach it to node 2 on C.2. Use the following AMI command for each link:

```
conf atmroute pnni interface modify <port> <vpi> -nodeix 2
```

Reboot C.2. This brings the link between C.2 and C.4 back to being operational.

12. Upon reboot of C.1, C.3 and C.4 will no longer need to be gateway switches. So, modify the default protocol to PNNI using the following AMI command on each switch:

    ```
    conf atmroute domain modify <domain ID> pnni
    ```

    Reboot both C.3 and C.4.

This completes the migration of peer group C to PNNI. Peer groups A and B can be migrated to PNNI in a similar way using the steps found in Section C.3.1.1.1.3. Once they are changed over to PNNI, the entire conversion is complete.

After the migration, the network has four areas at two discrete levels. Each area is a PNNI peer group. The final state of the network upon completion of the conversion is shown in Figure C.14.

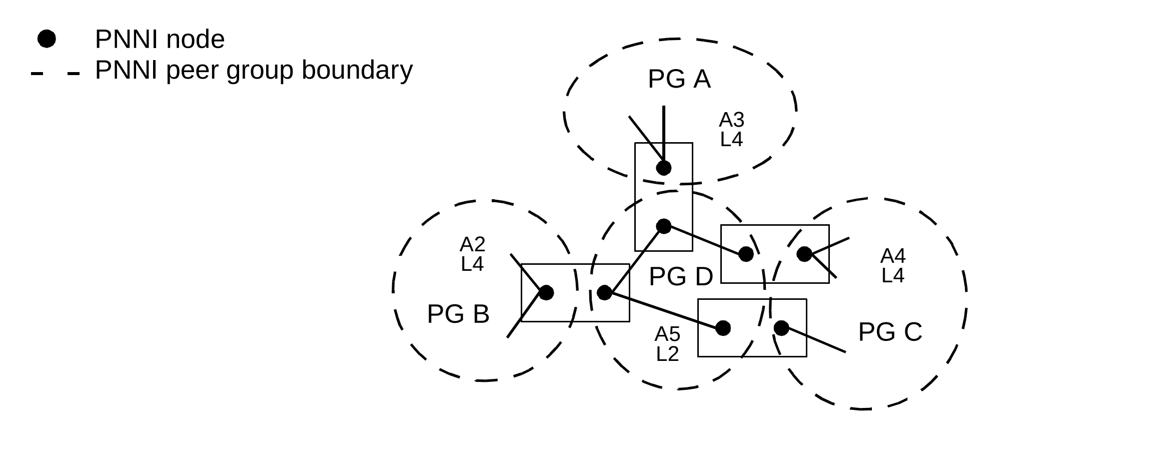

Figure C.14 - A Completely Converted PNNI Network

C.3.1.2 Migration Starting with the Peer Groups

This section gives a detailed example of how to migrate a hierarchical FT-PNNI network with a contiguous backbone as shown in Figure C.9 to a PNNI network. In that figure, only the border nodes are shown. The individual nodes within each peer group are not shown.

C.3.1.2.1 Overview of the Migration

The following basic steps are involved in the migration. Each of these steps is described in detail in the following sections. It is recommended that you read the entire section before attempting to change over your network.

1. Upgrade each switch in the network to *ForeThought* 6.0x.
2. Convert the individual peer groups of the FT-PNNI network.
3. Convert the backbone of the FT-PNNI network.

C.3.1.2.1.1 Upgrade the Switches

Upgrade all of the switches from to *ForeThought* 6.0.x using the following AMI command:

```
oper upgrade <remotehost>:<full path to remotefile>
```

For more information about upgrading your switches, see Chapter 4 of the *ATM Switch Installation and Maintenance Manual.*

C.3.1.2.1.2 Convert Peer Group C

1. Peer group C is the first to be converted. Start with switch C.6. Modify the default protocol of the default domain in C.6 to `gateway` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> gateway
   ```

   Reboot the switch. This creates the first PNNI node in the network with a default area 5 and level 5 as shown in Figure C.15.

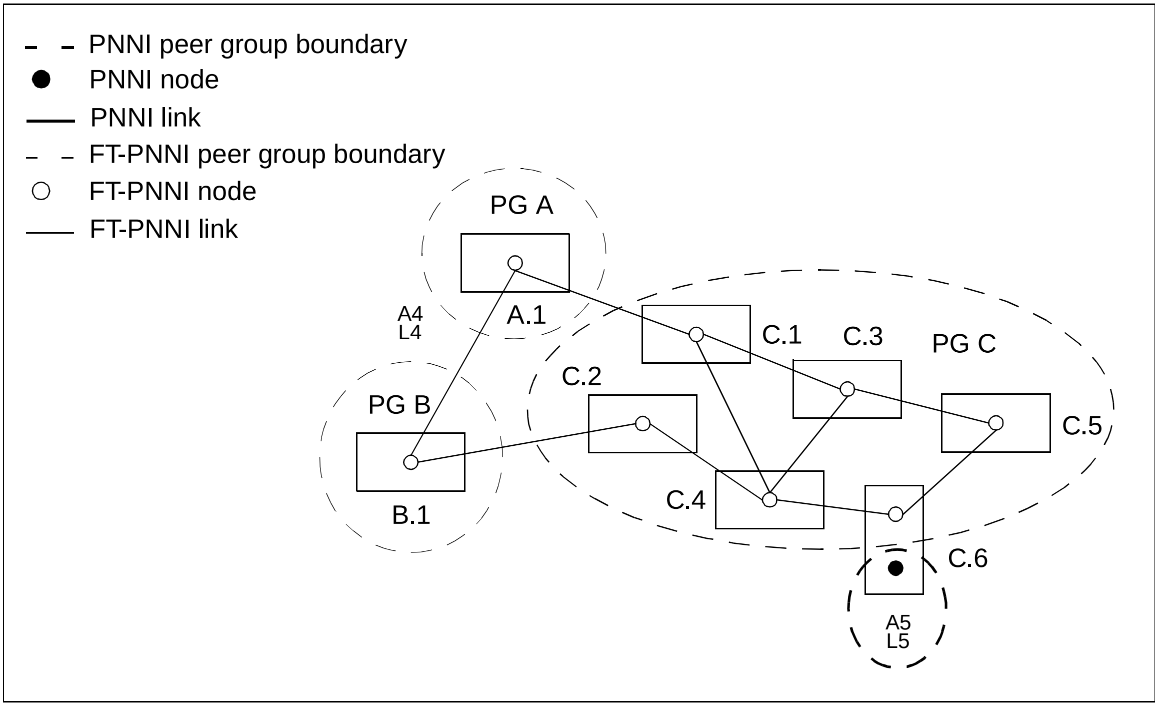

Figure C.15 - C.6 as a Gateway

2. Modify the default protocol of the default domain in C.5 to `gateway` using the following AMI command:

```
conf atmroute domain modify <domain ID> gateway
```

   Reboot switch C.5. This creates the second PNNI node in the network and the first PNNI link in area 5 between C.6 and C.5.

3. Modify the default protocol of the default domain in C.3 to `gateway` using the following AMI command:

```
conf atmroute domain modify <domain ID> gateway
```

   Reboot switch C.3. This creates the second PNNI link in the network between C.5 and C.3.

4. At this point, C.5 does not have any more FT-PNNI links left. So, modify the default protocol of the default domain in C.5 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.5.

5. C.4 is the next switch to be converted. Modify the default protocol of the default domain in C.4 to `gateway` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> gateway
   ```

   Reboot switch C.4.

6. Since C.6 has now lost its last FT-PNNI link, change the default protocol of the default domain in C.6 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.6.

7. C.1 is the next switch to be changed to a gateway. Since the FT-PNNI peer group of C will become partitioned when C.1 becomes a gateway switch, it is a good idea to change the peer group ID of the FT-PNNI node on C.1 so that other peer groups will not attempt to use peer group C as a transit peer group while computing inter-peer group routes. Change the peer group ID of the FT-PNNI node on C.1 to E (something other than C) using the following AMI commands:

   ```
   conf atmroute ftpnni pgmask <mask>
   conf atmroute ftpnni prefix <prefix>
   ```

   Answer **no** to the question of whether or not you want to the switch to be rebooted.

8. Modify the default protocol of the default domain in C.1 to `gateway` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> gateway
   ```

   Now reboot switch C.1. Two more PNNI links between C.1 and C.3 and between C.1 and C.4 are created.

9. Since C.3 no longer needs to be a gateway switch, modify the default protocol of the default domain in C.3 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.3.

10. C.2 is the last switch in the peer group to become a gateway. Modify the default protocol of the default domain in C.2 to `gateway` using the following AMI command:

```
conf atmroute domain modify <domain ID> gateway
```

    Upon the reboot of C.2, the last FT-PNNI link (between C.2 and C.4) in the peer group becomes a PNNI link.

11. Modify the default protocol of the default domain in C.4 to `pnni` using the following AMI command:

```
conf atmroute domain modify <domain ID> pnni
```

    Reboot switch C.4.

This completes the migration of peer group C to PNNI. The current state of the network is shown in Figure C.16.

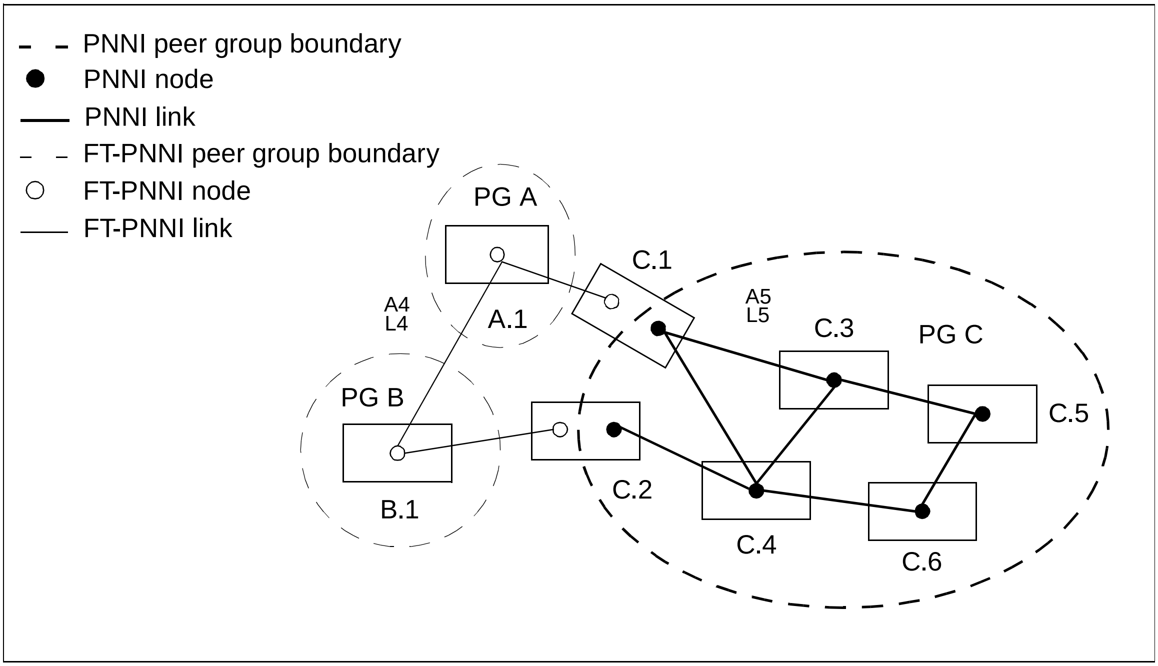

Figure C.16 - Peer Group C Fully Migrated to PNNI

C.3.1.2.1.3 Convert Peer Group A

Peer group A is chosen next to be migrated to PNNI. Step step 7 from the conversion of peer group C is unnecessary in peer group A’s case because there is only one border node in this peer group, and, therefore, the peer group does not get partitioned during the migration. Also, ensure that the area ID of the PNNI nodes in peer group A is different from the area ID of peer group C. Make 6 the area ID of peer group A and make the level 5 (the same level as peer group C).

Migrate all of the nodes in peer group A using the same method that was used to migrate peer group C in Section C.3.1.2.1.2.

C.3.1.2.1.4 Convert the Backbone

Once all of the nodes in peer group A have been modified, the link between A.1 and C.1 will become an inoperational PNNI link because it is between two gateway switches and because the peer group IDs of the PNNI nodes on A.1 and C.1 do not match. However, connectivity between the peer groups is still possible through the FT-PNNI backbone.

1. Modify the default protocol of the default domain in C.1 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.1.

2. Create a second PNNI node in C.1 with a node index of 2, the peer group ID set to D, area 3, and level 2 using the following AMI commands:

```
conf atmroute pnni node modify 2 -pgid d -forelevel 2 -forearea 3
```

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch C.1. This is the first node in the backbone area.

3. Modify the interface on C.1 corresponding to the link to A.1 and attach it to the newly-created node 2 on C.1 using the following AMI command:

```
conf atmroute pnni interface modify <port> <vpi> -nodeix 2
```

4. Change the default protocol of the default domain in A.1 to `pnni` using the following AMI command:

   ```
   conf atmroute domain modify <domain ID> pnni
   ```

   Reboot switch A.1. The FT-PNNI link between A.1 and B.1 is lost at this stage and connectivity between peer group A and the rest of the network is severed.

5. Create a second PNNI node in A.1 with a node index of 2, the peer group ID set to D, area 3, and level 2 using the following AMI commands:

```
conf atmroute pnni node modify 2 -pgid d -forelevel 2 -forearea 3

conf atmroute domain modify <domain ID> pnni
```

Reboot switch A.1.

6. Modify the interface on A.1 corresponding to the link to C.1 and attach it to the newly-created node 2 on A.1 using the following AMI command:

```
conf atmroute pnni interface modify <port> <vpi> -nodeix 2
```

At this point, the connectivity between peer groups A and C should be restored, but peer groups A and B still remain disconnected. The current state of the network is shown in Figure C.17.

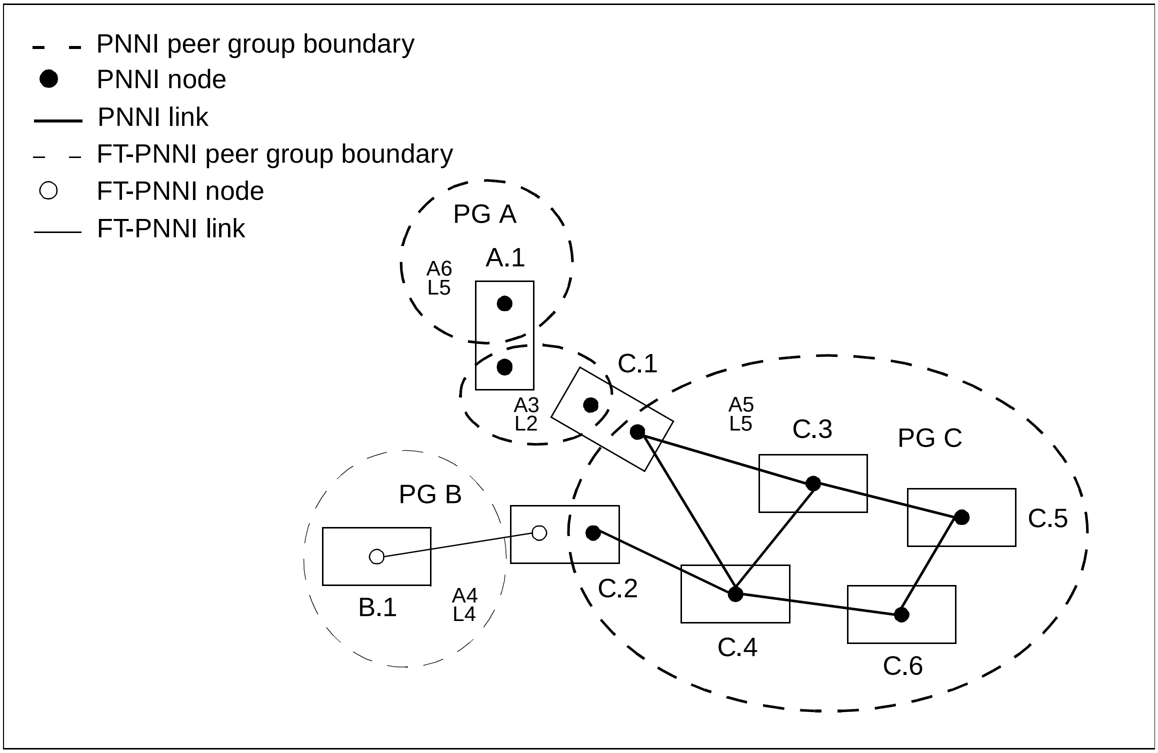

Figure C.17 - Peer Group B Disconnected from Peer Group A

To quickly restore the connectivity, the rest of the backbone should be migrated to PNNI.

7. Change the peer group ID to D, the level to 2, and area to 3 on the PNNI node in B.1, and change the default protocol of the default domain in B.1 to `gateway` using the following AMI commands:

```
conf atmroute pnni node modify 1 -pgid d -forelevel 2 -forearea 3
         conf atmroute domain modify <domain ID> gateway
```

   Reboot switch B.1. The link between A.1 and B.1 should now be restored as a PNNI link and connectivity resumes between A and B.

8. C.2 no longer needs to be a gateway switch. Modify the default protocol of the default domain in C.2 to `pnni` using the following AMI command:

```
conf atmroute domain modify <domain ID> pnni
```

   Reboot switch C.2. Upon the reboot, the backbone and peer groups A and C are now running PNNI and only peer group B needs to be migrated. The current state of the network is shown in Figure C.18.

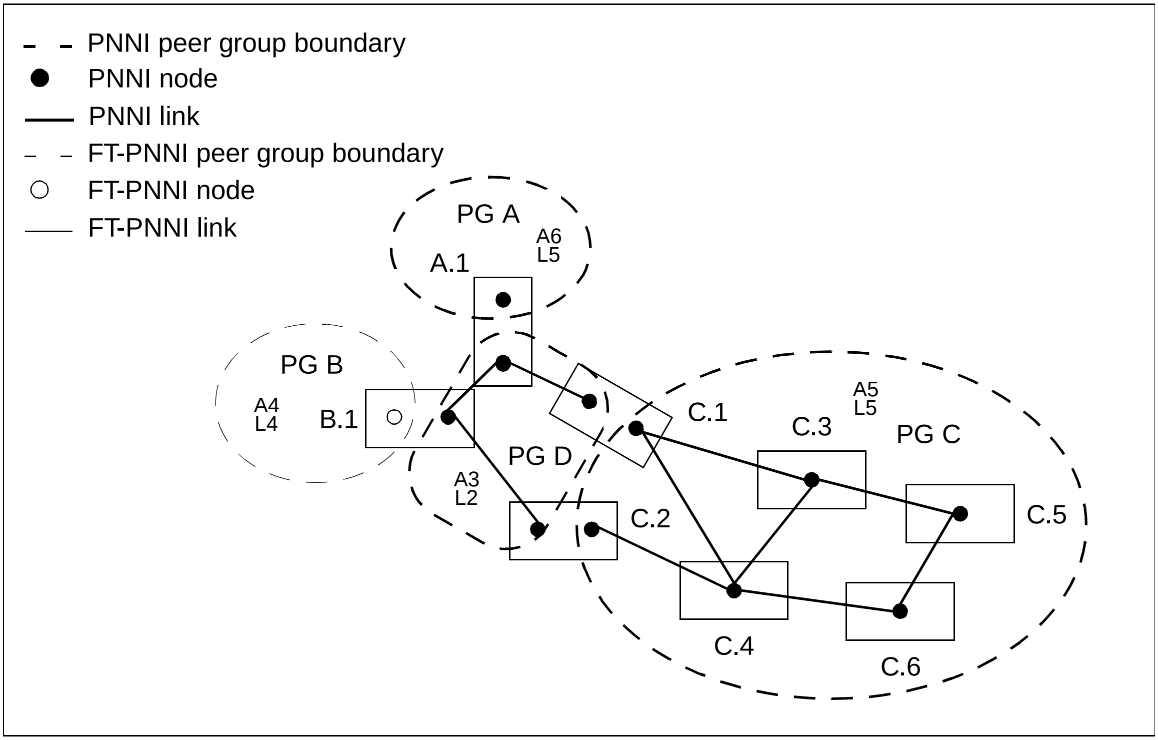

Figure C.18 - A Migrated Backbone

C.3.1.2.1.5 Convert Peer Group B

Migrate all of the nodes in peer group B using the same method that was used to migrate peer group C in Section C.3.1.2.1.2. Once peer group B has been converted, the migration to PNNI is complete.

C.3.2 Migration of a Hierarchical FT-PNNI Network with a Non-Contiguous Backbone

This section gives a detailed example of how to migrate a hierarchical FT-PNNI network with a non-contiguous backbone as shown in Figure C.19 to a PNNI network. In this figure, only the border nodes are shown. The individual nodes within each peer group are not shown.

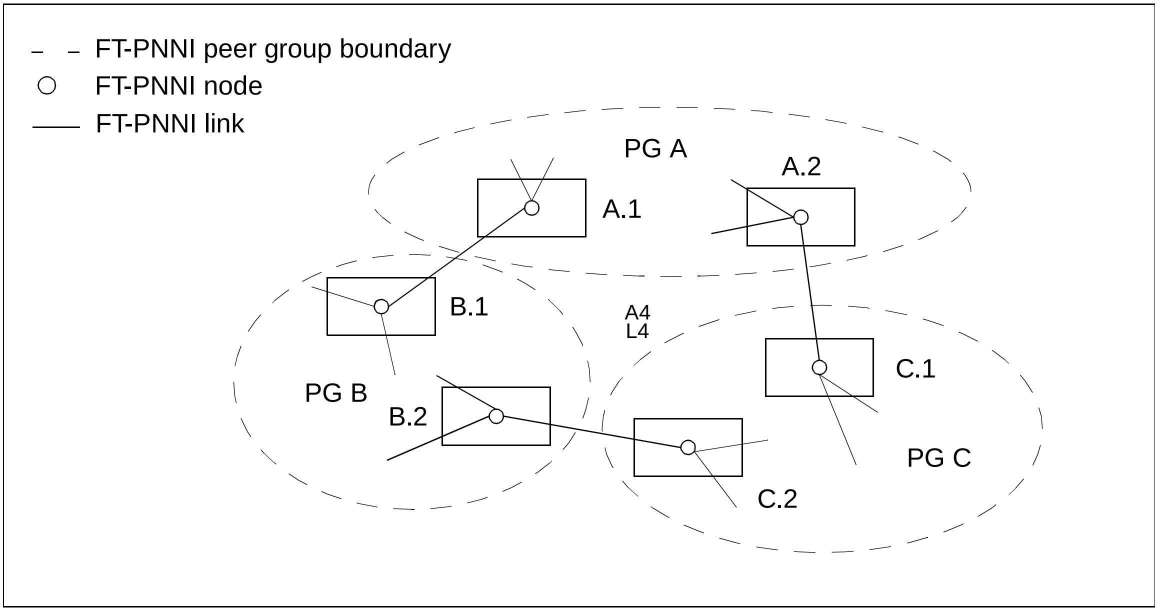

Figure C.19 - Hierarchical FT-PNNI Network with 3 Peer Groups and a Non-contiguous Backbone

Upon the complete migration of this network to PNNI, the network is modified as follows:

- Make peer group A the backbone area in a split-switch based hierarchical network.
- Peer group A would become a PNNI area with an area ID of 2 and level 2 with a single PNNI peer group in the new network.
- Peer group B would become a PNNI area with an area ID 1 and level 4 (a lower level than 2) with a single PNNI peer group.
- Peer group C would become a PNNI area with an area ID 3 and level 4 (a lower level than 2) with a single PNNI peer group.

- The link between B.2 and C.2 in the FT-PNNI network can be placed in a new area (with peer group ID D), which is a higher level area connecting peer groups B and C and acts as a back door entry between the two peer groups. This provides the redundancy that this link offered in the original FT-PNNI network.

Figure C.20 shows the state of the network on the completion of the migration.

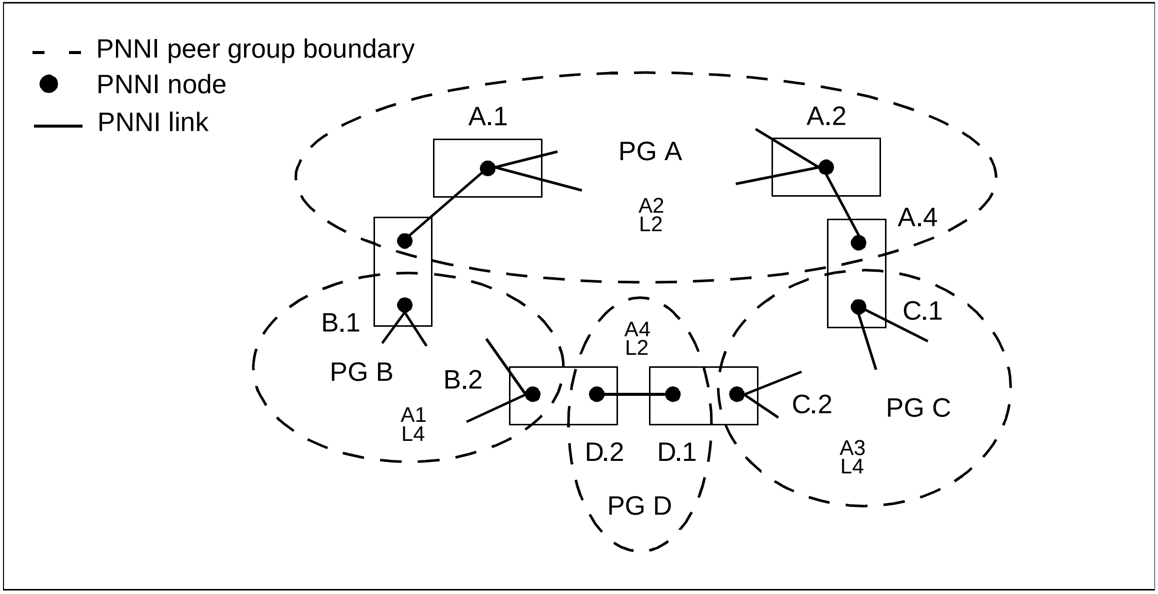

Figure C.20 - Hierarchical PNNI Network after Migration

C.4 Migration of a Non-hierarchical PNNI Network to a Lop-sided Hierarchical PNNI Network

This section discusses the conversion of a non-hierarchical (single peer group) PNNI network to a lop-sided hierarchical PNNI network. A lop-sided hierarchical network is one in which lowest level nodes and Logical Group Nodes (LGNs) are acting as peers in a single peer group. In this example, all switches in the network are FORE switches.

C.4.1 Migration Overview

This section gives a detailed example of how to change the non-hierarchical PNNI network shown in Figure C.21 to a lop-sided hierarchical PNNI network.

1. Upgrade each switch in the network to *ForeThought* 6.0.x.
2. Select a peer group leader (PGL) candidate. Set its node PGL priority to a non-zero value.
3. Configure a parent for that node.
4. Verify the election on all of the switches.
5. Change the interface on the node that resides in the higher level peer group.
6. Verify that this interface becomes an outside link.
7. Administer the de-activated node down.
8. Verify that the LGN has set up an SVCC RCC.
9. Repeat step 1 through step 8 for each peer group.

NOTE

PMP connections are not supported on an *ASX*-1000, *ASX*-1200, or *TNX*-1100 that is part of a multi-level hierarchy.

C.4.2 Detailed Migration Example

This section gives a detailed example of how to change the non-hierarchical PNNI network shown in Figure C.21 to a lop-sided hierarchical PNNI network.

NOTE

This migration procedure works only when addresses and address prefixes in the lower level peer groups (B.X and B.Y in Figure A.1) can be summarized by the higher level peer group (B in Figure A.1). If they are not summarized by the higher level peer group (e.g., if B.X was peer group F, B.Y was peer group G, and B was peer group H), the procedure can still be used, but an extra step is needed after step 5. The PG IDs of B.3 and B.2 in the higher level peer group (B) needs to be modified so that LGN B.X can reach those addresses.

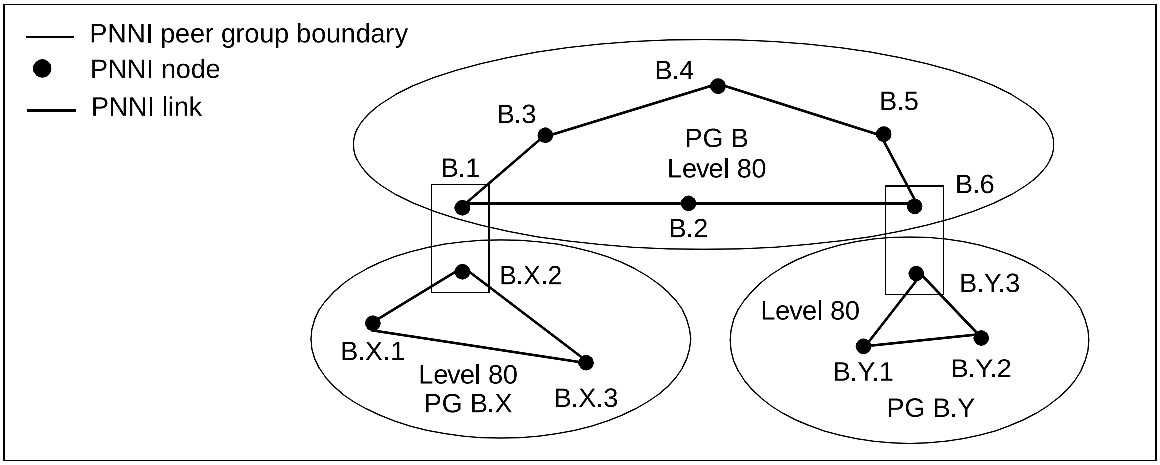

Figure C.21 - Non-hierarchical PNNI Example

1. Upgrade each switch in the network to *ForeThought* 6.0.x as follows:

   `oper upgrade` *`<remotehost>`*`:`*`<full path to remotefile>`*

   For more information about upgrading your switches, see Chapter 4 of the *ATM Switch Installation and Maintenance Manual.*

2. Start by migrating peer group B.X. Select a peer group leader (PGL) candidate and set its node PGL priority to a non-zero value so that it gets elected as PGL. In this example, B.X.1 is used.

```
conf atmr pnni node modify 1 -pglprio 5
```

3. Configure a parent for that node using the following command:

```
conf atmr pnni node newparent <parentix> <childix> -level <level>
```

```
conf atmr pnni node newparent 2 1 -level 72
```

4. Verify the election on all of the switches in peer group B.X using the following command:

```
conf atmr pnni node show pgl
```

On the node that has been elected PGL, the `PglState` should say `operPgl`. At this point, an LGN B.X representing peer group B.X has been activated by the PGL B.X.1 as shown in Figure C.22.

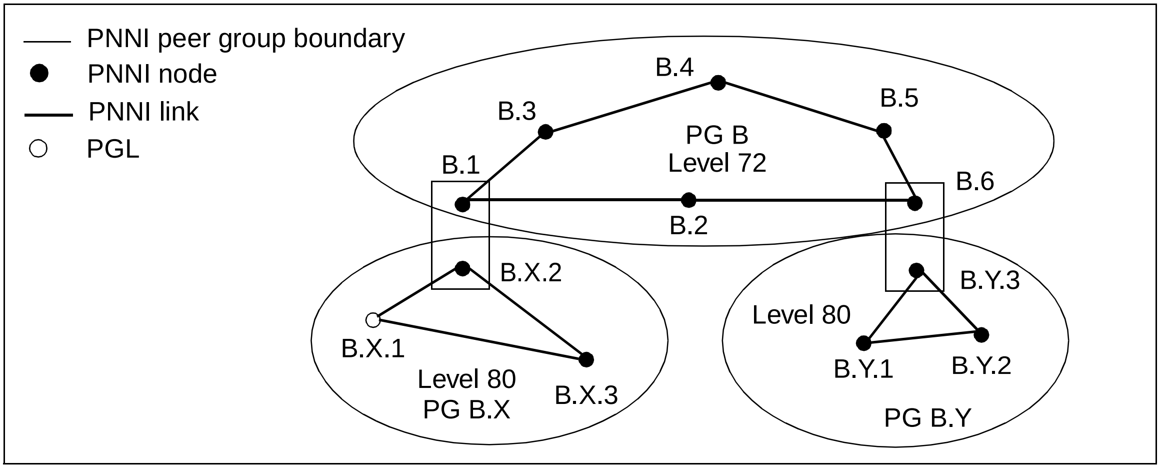

Figure C.22 - B.X.1 Elected as PGL

5. On the split switch containing B.X.2 (lower level node) and B.1 (higher level node), move any links attached to the higher level node (B.1) to the lower level node (B.X.2) as shown in Figure C.23. Change the interfaces on node 1 (B.X.2) using the following commands:

```
conf atmr pnni interface modify 1C1 0 -nodeix 1

conf atmr pnni interface modify 1D1 0 -nodeix 1
```

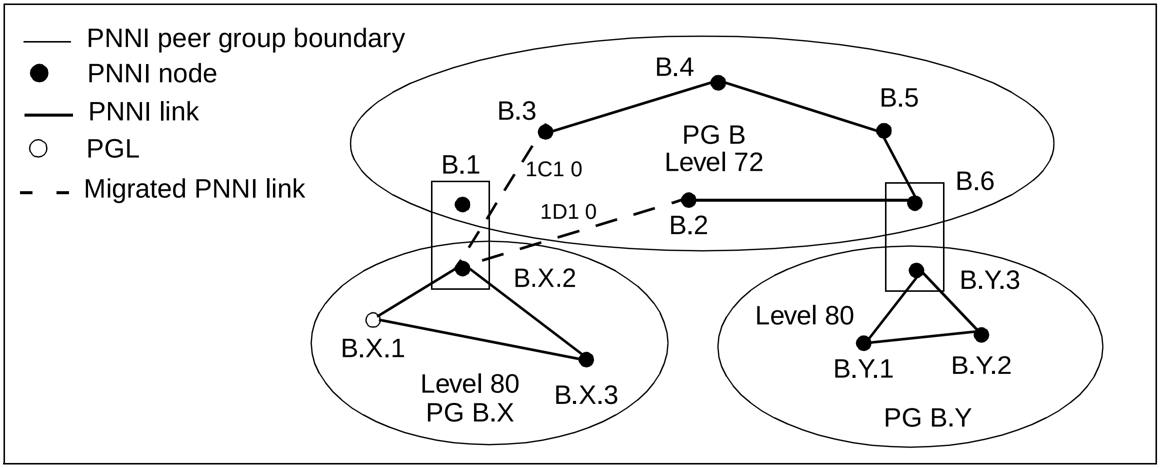

Figure C.23 - Migrating the Links

6. These links become PNNI outside links because they connect nodes in two different peer groups (PG B.X and PG B). Verify that 1C1 0 and 1D1 0 become outside links by noting that the `HelloSt` for each says `twoWayOutside` using the following command:

```
conf atmr show
```

7. Node B.1 (node 2 on the split switch) can now be de-activated. Administer this node down as follows:

```
conf atmr pnni node admin 2 down
```

The peer groups now look like the example in Figure C.24.

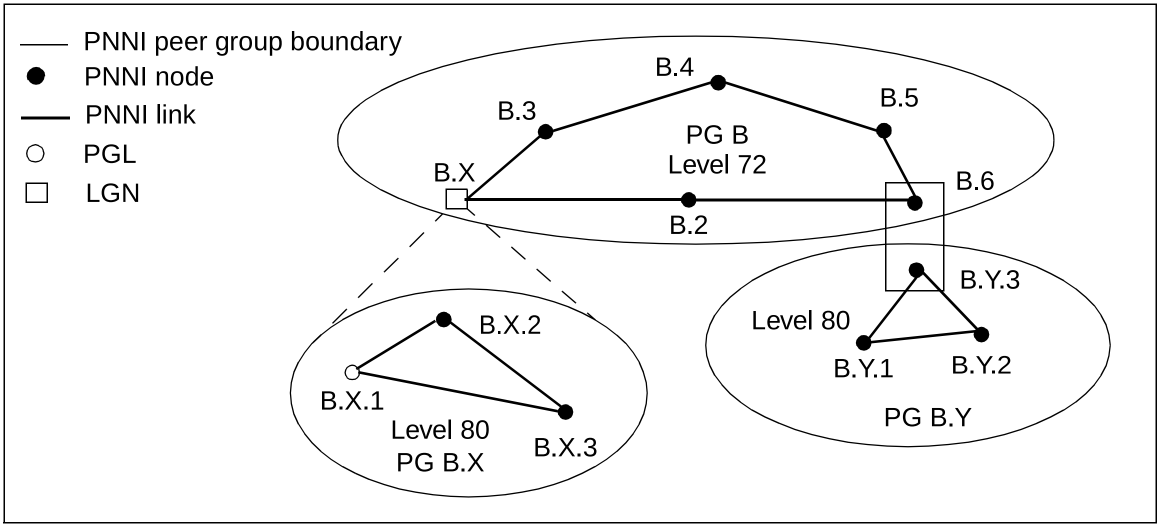

Figure C.24 - Node B.1 is De-activated and LGN B.X Represents Peer Group B.X

8. LGN B.X (residing at the same switch as B.X.1) now sets up SVCC RCCs to B.2 and B.3 and initiates a peer relationship with them. LGN B.X now becomes an integral part of peer group B. Verify that the LGN has set up an SVCC RCC with B.2 and B.3 as follows:

   a. On the LGN switch, use the following command:

   ```
   display atmr pnni svccrcc
   ```

   The `HelloState` should say `twoWayInside` because the switches are within the same peer group.

   b. On switches B.2 and B.3, use the following command:

   ```
   display atmr pnni svccrcc
   ```

   The `HelloState` should say `twoWayInside` because the switches are within the same peer group.

9. Repeat step 1 through step 8 on peer group B.Y to convert it in a similar manner so that B.Y.3 is the PGL in peer group B.Y and so that LGN B.Y is created as shown in Figure C.25.

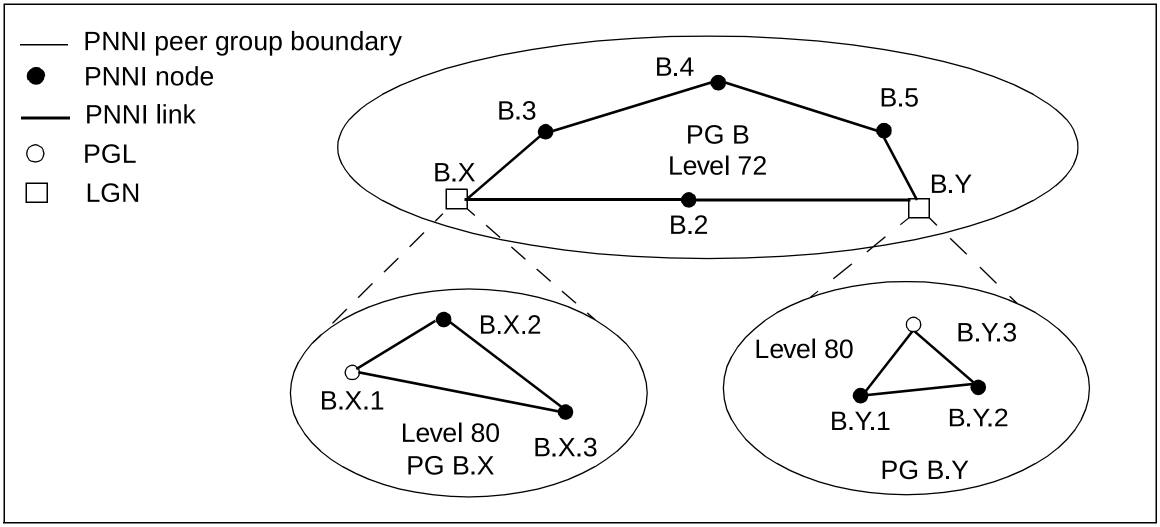

Figure C.25 - Completed Conversion

The migration is now complete.

APPENDIX D Configuring *FramePlus* Modules

FORE Systems' *FramePlus*™ network modules (NMFR-4/DS1A and NMFR-4/E1A) are four-port interfaces that support interworking between ATM and Frame Relay, or ATM and Frame-based UNI (FUNI).

All four ports of the DS1 *FramePlus* Network Module can support fractional DS1 services (n x 64 Kbps) where 1 to 24 contiguous or non-contiguous DS0 channels are mapped to a single ATM port.

All four ports of the E1 *FramePlus* Network Module can support fractional E1 services (n x 64 Kbps) where 1 to 31 contiguous or non-contiguous DS0 channels are mapped to a single ATM port.

This appendix provides an overview of a *FramePlus* network module and provides configuration examples. Topics that are covered include the following:

D.1 Frame Relay Overview

Serving as an interface between user and network equipment, Frame Relay provides a means for statistically multiplexing many logical data channels (or virtual circuits) over a single physical transmission link. Frame Relay is ideal for supporting multiple data streams because it provides flexible and efficient utilization of available bandwidth.

D.1.1 Interworking Function (IWF)

When connecting an ATM Customer Premise Equipment (CPE) and a Frame Relay CPE, each device has no knowledge that the distant device supports a different protocol. The *FramePlus* network module performs a service interworking function (IWF) to terminate the respective Frame Relay and ATM services and to perform appropriate interworking conversions before transmitting the CPE information. IWF uses two different modes of operation to address interoperability between upper layer user protocol encapsulation techniques on the *FramePlus* network module: translation mode and transparent mode.

D.1.1.1 Translation Mode

When the two CPEs support different standard encapsulation protocols, a translation, or re-encapsulation, must take place in the CPE payload portion of the cells that are being transmitted. Because Frame Relay supports RFC 1490 encapsulation and ATM supports RFC 1483 encapsulation, the information in the frames is converted to cells and vice versa.

D.1.1.2 Transparent Mode

When both CPEs support the same encapsulation protocol, there is no need for any translation or re-encapsulation of the CPE payload portion of the cells that are being transmitted. Therefore, the traffic is sent “transparently.” Transparent mode must be used if the CPEs support non-standard encapsulation protocols. When the transparent method of IWF is used, there is a mapping that occurs of each Frame Relay DLCI to an ATM VPI, VCI combination.

One example would be if proprietary encapsulation techniques are being used. A second example would be if there is FR/CPE equipment at both edges of the network with ATM providing the transport, there would be no need for any translation.

D.2 Configuring the Module

Before configuring any services or PVCs, you should divide the buffer space between the high and low priority buffers as described in Section D.2.1. Cells are read from the high priority buffer first. After the high priority buffer is emptied, cells are read from the low priority buffer. Then you should set the EPD/PPD thresholds for the high and low priority buffers as described in Section D.2.2.

D.2.1 Dividing the Buffer Space

The FramePlus network module has two priority buffers: high and low. The total, combined buffering space available is 32,768 cells. The amount of buffering space is split between the two buffers using **conf module fram setmem**. There are four fixed configuration models for partitioning the buffer sizes as shown in Table A.1. The default model is highzero.

Table D.1 - Buffer Models

Model	High Priority Buffer	Low Priority Buffer
highzero	0 cells	32,768 cells
high1quarter	8,192 cells	24,576 cells
high2quarter	16,384 cells	16,384 cells
high3quarter	24,576 cells	16,384 cells

If you choose highzero, you want all of the buffering space to go in the low priority buffer and none in the high priority buffer. If you choose high1quarter, then one fourth of the buffering space goes in the high priority buffer and three fourths goes in the low priority buffer, and so on.

Before you can change the buffer allocation, you must first take the network module out of service by administering it down as follows:

```
myswitch::configuration module> admin 1d down
Disabling the network module will destroy all
existing connections going through it.

Disable the network module [n]? y

myswitch::configuration module> fram
myswitch::configuration module fram> set 1d high2quarter
```

The network module comes back up after you make the change so you do not need to use the **conf module admin** *<module>* **up** command.

Configuring FramePlus Modules

D.2.2 Setting the Thresholds

After the buffer space is configured, set the various discard thresholds under **`conf module fram`**.

NOTE

It is important that you follow the order shown in this section for configuring the thresholds. The calculation of each threshold is dependent on the previous one.

For this buffer configuration example, assume you have set the buffers to use high2quarter, meaning that both the high and low priority buffers have 16, 384 cells as shown in Figure D.1.

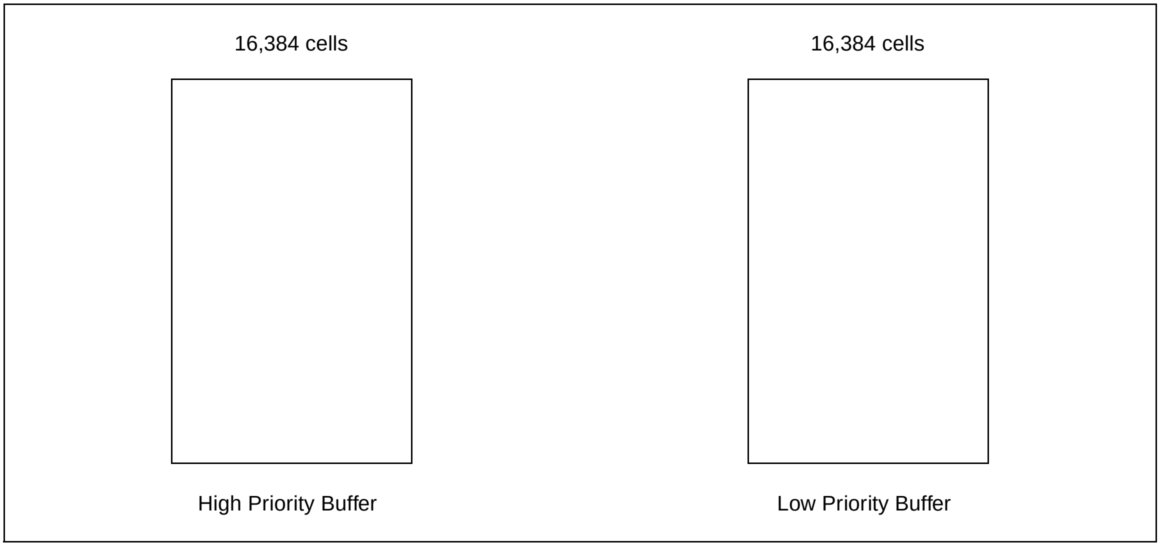

Figure D.1 - Buffer Sizes Configured Using the setmem Command

NOTE

The number of cells shown in this section for buffer sizes are approximations.

D.2.2.1 Noting the CLP0PPD Threshold

Each buffer has four different thresholds. The first is for Partial Packet Discard (PPD) on cells with CLP=0. This threshold is automatically set for you as 87.5% of the size of each buffer and cannot be changed. (It is not displayed in AMI currently, but you need to know this value because the other thresholds are calculated on the remaining buffer size.)

Therefore, in this example, since the total buffer size you set is 16,384 cells then the threshold for CLP0PPD is 87.5% of each buffer as shown in Figure D.2:

16,384 cells * 0.875 percent = 14,336 cells

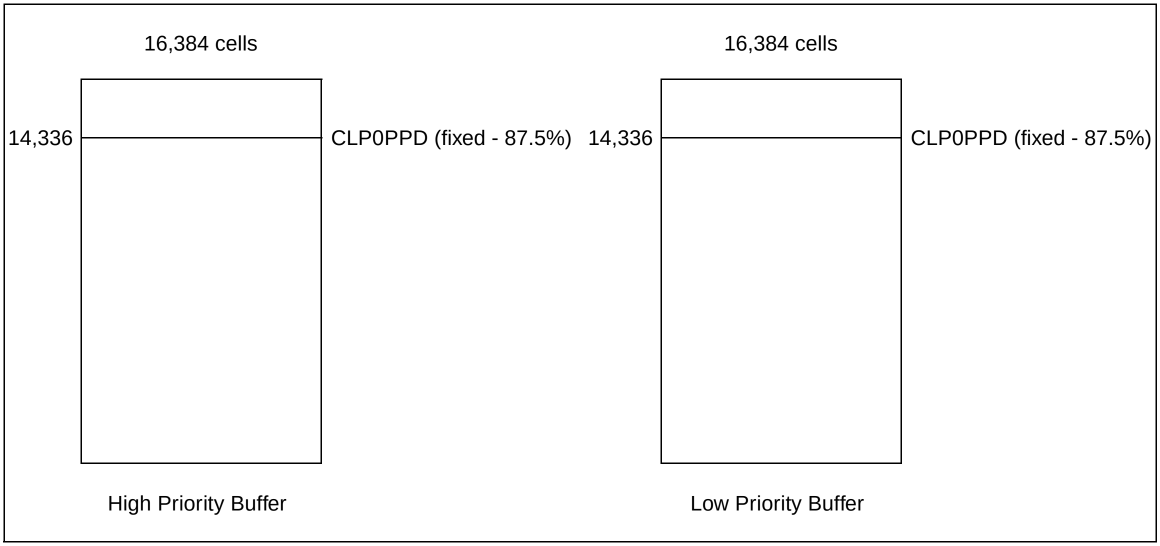

Figure D.2 - CLP0PPD Automatically Calculated

D.2.2.2 Configuring the CLP1EPD Threshold

Next, calculate the CLP1EPD threshold for each buffer. You have four choices: 25, 37, 50, or 62 percent. For example, you could set the threshold to 50 percent of the remaining buffer size of each buffer, which is 14,336, as shown by the arrow in Figure D.3:

14,336 cells * 0.5 = 7,168 cells

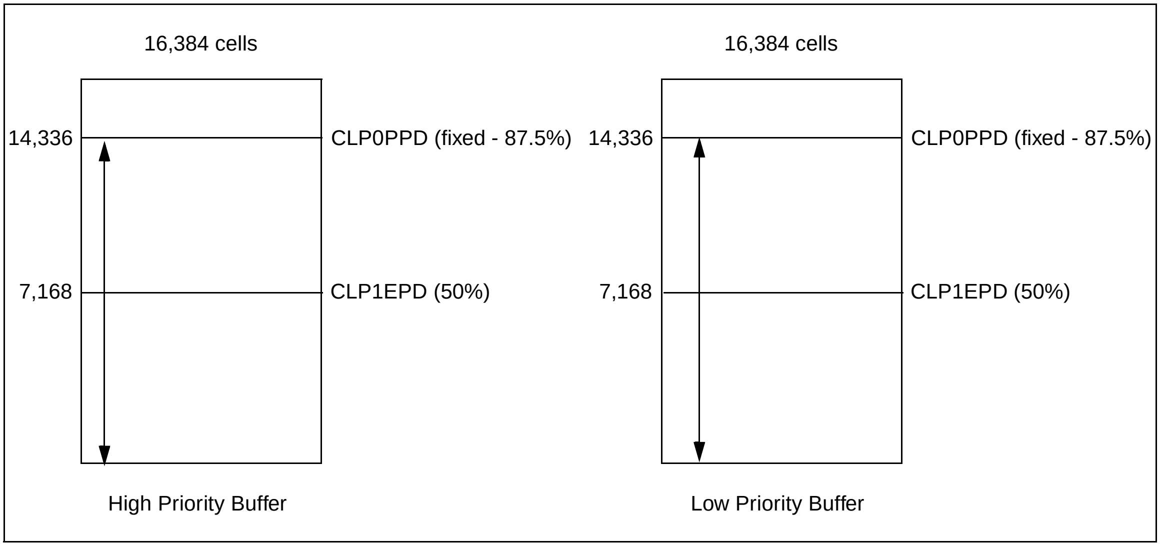

Figure D.3 - Calculated CLP1EPD

D.2.2.3 Configuring the CLP0EPD Threshold

Now, calculate the CLP0EPD threshold for each buffer. You have four choices: 50, 62, 75, or 87 percent. For example, you could set the threshold to 50 percent of the remaining buffer size of each buffer, which is 7,168, as shown by the arrow in Figure D.4. Then, add the base of 7,168 back in.

7,168 cells * 0.5 = 3,584 cells + 7,168 cells = 10,752 cells

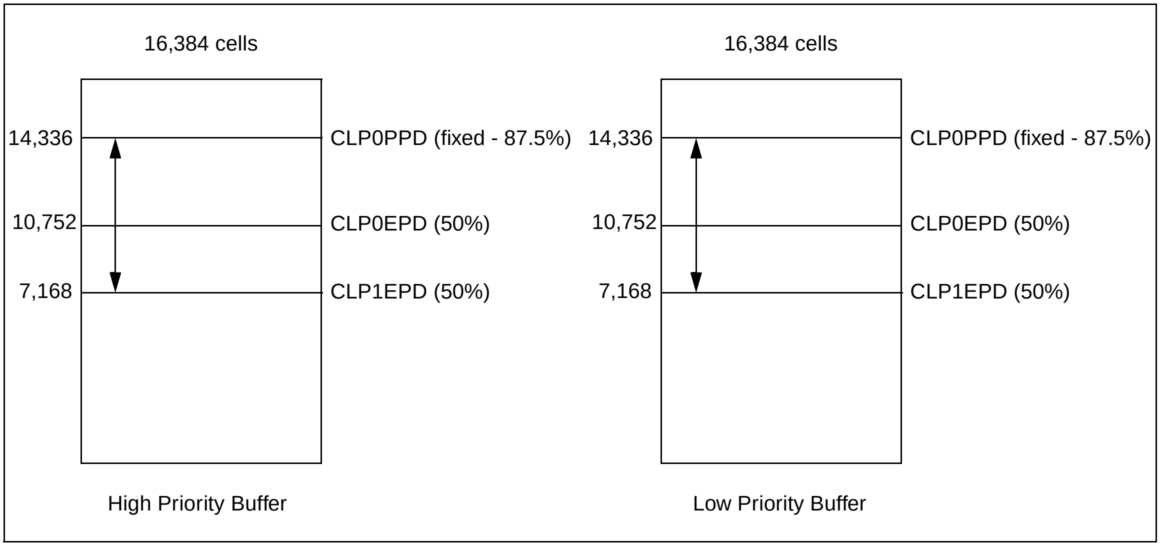

Figure D.4 - Calculated CLP0EPD

D.2.2.4 Configuring the CLP1PPD Threshold

Finally, calculate the CLP1PPD threshold for each buffer. You have four choices: 50, 62, 75, or 87 percent. For example, you could set it to 50 percent of the remaining buffer size, which is 3,584, as shown by the arrow in Figure D.5. Then, add the base of 7,168 back in as shown:

10,752 cells - 7,168 cells = 3,584 cells

3,584 cells * 0.5 = 1,792 cells

1,792 cells + 7,168 cells = 8,960 cells

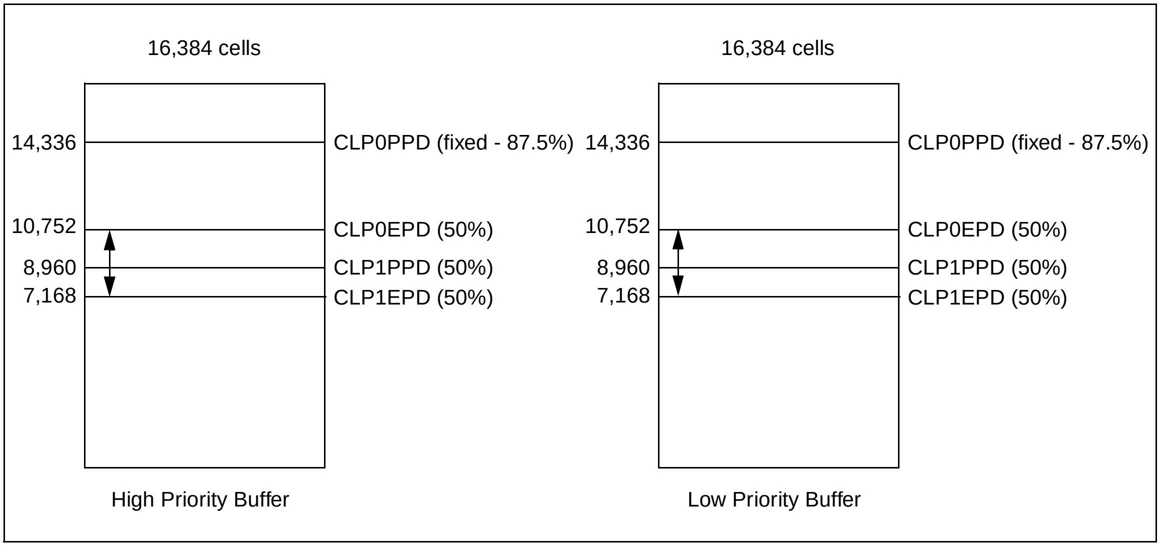

Figure D.5 - Calculated CLP1PPD

D.3 Profiles

A profile lets you define a set of information that can be applied to a particular service or PVC using a single index number, much in the same way that you can define a set of traffic management information in a UPC contract and then apply it to a PVC using a single index number.

For most profile types, several default profiles are provided by the switch for your convenience. These default profiles reflect what typical networks may need. You cannot delete the default profiles. If a service or connection is created without specifying any profile, the first default profile in the list (index 0) is used.

You can also create new profiles and delete existing profiles, but you cannot modify existing ones. Each profile has an associated reference count, which identifies the number of services or connections that are currently using that profile. You cannot delete any profile that is currently in use (has a non-zero reference count in the `Ref` field under the profile's **`show`** command).

There are six different profiles that are supported by the *FramePlus* network module: FRF.8, EPD/PPD, FRRATE, LMI, service, and FUNI. The FRF.8, and FRRATE profiles can only be applied to Frame Relay PVCs. The FUNI profiles can only be applied to FUNI PVCs. The EPD/PPD profiles can be applied to either Frame Relay or FUNI PVCs. The LMI profiles can only be applied to Frame Relay services. The service profiles can be applied to either Frame Relay or FUNI services. All profiles can be configured under the **`conf profile`** menu. Please see Part 2 of the *AMI Configuration Commands Reference Manual* for detailed information about the AMI parameters. Each of these profiles is discussed in the following sections.

D.3.1 EPD/PPD Profile

The Early Packet Discard/Partial Packet Discard (EPD/PPD) profile lets you determine how selective cell discard is performed. The profile then can be applied on a per-VC basis to a Frame Relay PVC using the **`-epdppd`** *`<index>`* option under **`conf fratm pvc new`** or to a FUNI PVC using the **`-epdppd`** *`<index>`* option under **`conf funi pvc new`**. The EPD/PPD profile lets you specify which priority buffer (high or low) will handle the traffic of the given PVC. The different thresholds can be enabled or disabled per PVC. These thresholds can be configured under **`conf module fram`**.

D.3.2 FRF.8 Profile

FRF.8 is the Frame Relay Forum's FR/ATM PVC Service Interworking Implementation Agreement. The FRF.8 profile allows you to define the interworking mappings between Frame Relay and ATM. The profile then can be applied on a per-VC basis to a Frame Relay PVC using the **`-frf8`** *`<index>`* option under **`conf fratm pvc new`**.

The FRF.8 profile lets you decide if the Discard Eligibility (DE) field in every frame is mapped to the Cell Loss Priority (CLP) field in every ATM cell. It lets you define if the Forward Explicit Congestion Notification (FECN) field in every frame is mapped to the EFCI field of every ATM cell. It also lets you define the possible encapsulation translation protocols that can be used.

D.3.3 Frame Relay Rate Profile

The Frame Relay rate profile allows you to manage and define rate enforcement and rate adaptation characteristics that can be applied on a per-VC basis to a Frame Relay PVC using the **`-frrate`** *`<index>`* option under **`conf fratm pvc new`**.

You can explicitly enable rate enforcement in both the ingress (Frame Relay to ATM) and the egress (ATM to Frame Relay) direction on a *FramePlus* network module. Ingress rate enforcement is configured using the parameters **`inBc`**, **`inBe`**, and **`inCir`**. Egress rate enforcement is configured using the parameters **`outBc`**, **`outBe`**, and **`outCir`**. The rate adaptation parameters are **`minBc`** and **`cmPeriod`**. If no rate enforcement parameters are specified, the default values are applied.

The egress rate enforcement also provides an option by which frames on a per-VC basis can be buffered as opposed to being discarded when thresholds have been crossed. This buffering option can increase goodput. The selection of buffering is configured at a service level, but only takes effect on a connection if egress rate enforcement has been enabled. The amount of buffering assigned to a connection is finite and is determined internally by the system. If the buffering has been selected, then the system holds traffic in excess of Bc+Be for a period of Tc (= Bc/CIR), before trying to resend the buffered traffic. If the buffer becomes full, excess traffic is discarded. The rate adaptation mechanism is activated for a PVC after an ingress frame, addressed to the PVC, has been received with BECN set.

The Frame Relay rate profile performs the exact same function that UPC contracts do for ATM. If you want to use the overbooking option in the service profile, then you need to apply a Frame Relay rate profile. Bc is the committed burst, Be is the excess burst, and CIR is the committed information rate. CIR is derived as follows:

$$\text{CIR} = \text{Bc} \; / \; \text{time period}$$

For example, if the Bc is 10 and the time period is 125 microseconds, then the CIR is 10 x 8192 or 81,920 bps.

EIR is the excess information rate and is derived as follows:

$$\text{EIR} = \text{Be} \; / \; \text{time period}$$

For example, if the Be is 10 and the time period is 125 microseconds, then the EIR is 10 x 8192 or 81,920 bps.

CIR is very similar to VBR SCR and EIR is very similar to VBR PCR.

D.3.4 LMI Profile

The Link Management Interface (LMI) profile lets you define the version of LMI that is supported by a particular Frame Relay service. The profile can be applied on a per-service basis using the **-lmi** *<index>* option under **conf fratm new**. If you do not want to use LMI, then enter the **none** option for **-flavor** under **conf fratm new** and use this profile when creating your services.

LMI serves a function that is similar to ILMI for ATM. LMI signalling occurs on DLCI 1023. LMI timers poll the network and report if services are up or down. This information can be optionally translated into ATM OAM F5 cells so that when a DLCI goes down, it triggers an F5 cell for the corresponding PVC. This profile lets you specify the values of the various LMI timers. Please see Part 2 of the *AMI Configuration Commands Reference Manual* for the default values and the ranges for the timers.

D.3.5 Service Profile

The generic service profile lets you specify service attributes that are common to Frame Relay interworking services and FUNI services. The profile then can be applied on a per-service basis to a Frame Relay service using the **-service** *<index>* option under **conf fratm new** or to a FUNI service using the **-service** *<index>* option under **conf funi new**.

This profile is important when using multiple timeslots on a port because you must ensure that the access rate of the referenced service profile index is large enough to support them. For example, the default access rate is 64 Kbps. If you leave the rate at 64, you must only use a single timeslot. If you set the rate to 128 Kbps, you must use two timeslots. If you set the rate 1536 Kbps, you must use 24 timeslots, etc.

This profile also lets you define the maximum frame size, the maximum number of DLCIs/PVCs supported, and input and output bandwidth overbooking for a given service.

D.3.6 FUNI Profile

The FUNI profile lets you determine the VPI, VCI range to use for FUNI connections. The profile then can be applied on a per-service basis to a FUNI service using the **-funi** *<index>* option under **conf funi new**.

D.4 Services

A service is a grouping of timeslots on a port. In this respect, a service is similar to an ATM PVP. Multiple DLCIs (connections) can exist within each service.

D.5 Configuring Frame Relay

To configure Frame Relay on a *FramePlus* network module, these steps must be performed in the following order:

1. Choose or create any profiles that you wish to use.
2. Create each service that you need and associate the profile(s) with it.
3. Create each PVC and associate the profile(s) and service(s) with it.

These steps are described in detail in the following sections. There are also steps described for configuring SPANS SPVCs and PNNI SPVCs on a *FramePlus* network module.

NOTE

A *FramePlus* network module can run either a Frame Relay over ATM or Frame-based User to Network Interface (FUNI) application. Because Frame Relay is the default, it is assumed that you do not need to change the application from FUNI to Frame Relay at this point. If you need to change the application, see the instructions in Section D.6.1. The `Appln` field under **conf mod show** displays which application is currently running (either `fratm` for Frame Relay or `funi` for FUNI).

D.5.1 Choosing Frame Relay Profiles

First, you need to decide which profiles you want to use. For most profile types, several default profiles are provided for your convenience. If a service or connection is created without specifying any profile, the first default profile in the list (index 0) is used.

To display the default profiles, use the show command under each profile; e.g., **conf profile epdppd show**. If these defaults do not match your network's needs, you can create your own using the **new** command under each profile; e.g., **conf profile epdppd new**. Create the new profiles for each profile type that you are going to use. Once you have determined which default profiles to use, you can move on to Section D.5.2.

D.5.2 Creating the Services for Frame Relay

Create any Frame Relay services that you want and apply the LMI and service profiles. For example:

```
myswitch::configuration fratm> new <port> <timeslots> [-lmi <index>]
[-service <index>] [-egress_re (enable|disable)] [-admin (up|down)]
[-name <name>] [-id <bnp:cc>]

myswitch::configuration fratm> new 4a1 1 -name service_a
The newly created service id is 4A1:00

myswitch::configuration fratm> new 4a1 2 -lmi 1 -name service_b
The newly created service id is 4A1:01

myswitch::configuration fratm> new 4a1 3 -name service_c
The newly created service id is 4A1:02

myswitch::configuration fratm> show
Searching for FR-ATM services
 SvcId  TimeSlot    Admin EgressRE Lmi Serv Traps    Stats    Name
 4A1:00 1           up    disabled 0   0    enabled  disabled service_a
 4A1:01 2           up    disabled 1   0    enabled  disabled service_b
 4A1:02 3           up    disabled 0   0    enabled  disabled service_c
```

As you can see in the `Stats` field, the collection of statistics is disabled, by default, at the service level. It is enabled, by default, at the module level. If you want to collect statistics, you must enable that functionality here at the service level as well:

```
myswitch::configuration fratm> stats <serviceid> (enable|disable)

myswitch::configuration fratm> stats 4A1:00 enable
myswitch::configuration fratm> stats 4A1:01 enable
myswitch::configuration fratm> stats 4A1:02 enable

myswitch::configuration fratm> show
Searching for FR-ATM services
 SvcId  TimeSlot    Admin EgressRE Lmi Serv Traps    Stats    Name
 4A1:00 1           up    disabled 0   0    enabled  enabled  service_a
 4A1:01 2           up    disabled 1   0    enabled  enabled  service_b
 4A1:02 3           up    disabled 0   0    enabled  enabled  service_c
```

D.5.3 Creating Frame Relay PVCs

Now that the profiles and services have been created, you can create your PVCs and apply the EPD/PPD, FRF.8, and FRRATE profiles to the PVCs. For example:

```
myswitch::configuration fratm pvc> new <serviceid> <dlci> [-oport <oport>]
       [-ovpi <ovpi>] [-ovci <ovci>] [-faupc <index>] [-afupc <index>]
       [-epdppd <index>] [-admin (up|down)] [-frrate <index>] [-frf8 <index>]
       [-name <name>]

note: if oport/ovpi/ovci aren't specified, a dangling FRATM PVC will be created.
```

NOTE

Currently, the *FramePlus* network module does not permit the establishment of Frame-to-Frame connections across a single fabric. This restriction prevents connections between *FramePlus* ports located on the same network module, or *FramePlus* ports located across separate modules, but attached to the same switch fabric.

```
myswitch::configuration fratm pvc> new 4A1:00 100 -oport 4c1 -ovpi 0 -ovci 100
-epdppd 1 -name pvc_a

myswitch::configuration fratm pvc> new 4a1:01 101 -oport 4c1 -ovpi 0 -ovci 101
-name pvc_b

myswitch::configuration fratm pvc> new 4a1:02 102 -oport 4c1 -ovpi 0 -ovci 102
-name pvc_c

myswitch::configuration fratm pvc> show
             Input          Output
 SvcId   dlci Port   VPI  VCI  Port VPI VCI Admin  Epd Frr Frf8 IngRE     Name
 4A1:00  101   4A1  16   32   4C1   0 101    up   1   0    0   disable  pvc_a
 4A1:00  101   4C1   0 101   4A1  16  32    up   1   0    0   disable  pvc_a
 4A1:01  100   4A1   0   32   4C1   0 100    up   0   0    0   disable  pvc_b
 4A1:01  100   4C1   0 100   4A1   0  32    up   0   0    0   disable  pvc_b
 4A1:02  102   4A1  32   32   4C1   0 102    up   0   0    0   disable  pvc_c
 4A1:02  102   4C1   0 102   4A1  32  32    up   0   0    0   disable  pvc_c
```

D.5.4 Configuring Frame Relay SPVCs

You can also configure SPVCs (Smart Permanent Virtual Circuits) on a *FramePlus* network module. An SPVC is a connection that spans multiple switch fabrics and looks like a PVC at the local and remote endpoints with an SVC in the middle. If a link carrying an SPVC goes down and there is an alternate route, then the end switch fabrics of the SPVC automatically reroute the SPVC around the failed link.

However, the procedure for creating both SPANS SPVCs and PNNI SPVCs on a *FramePlus* network module is slightly different than it is for other network modules. Section D.5.4.1 describes how to create a Frame Relay SPANS SPVC. Section D.5.4.2 describes how to create a Frame Relay PNNI SPVC.

D.5.4.1 Creating a Frame Relay SPANS SPVC

To create a Frame Relay SPANS SPVC, perform the following steps:

1. Configure the ingress (Frame Relay to ATM) Frame Relay PVC without specifying the **-oport**, **-ovpi**, and **-ovci** parameters. For example:

```
myswitch::configuration fratm pvc> new 4A1:00 100 -epdppd 1 -name spvc_a
```

2. Display the Frame Relay PVC so you can see what `Input Port`, `Input VPI`, and `Input VCI` values were assigned to the PVC. For example:

```
myswitch::configuration fratm pvc> show
             Input          Output
 SvcId   dlci Port   VPI VCI  Port VPI VCI Admin  Epd Frr Frf8 IngRE     Name
 4A1:00  100   4A1    0  32                 up    1   0    0   disable  spvc_a
```

3. Configure a new SPANS SPVC where the *<port>*, *<vpi>*, and *<vci>* values are the `Input Port`, `Input VPI`, and `Input VCI` values that are displayed in step 2. The `dest` parameters identify the endpoint of the connection. For example:

```
myswitch::> open 198.29.22.46 private
Opening a session for “198.29.22.46”, please wait...
Connected to “198.29.22.46” (asx200bx).

*fishtank::> localhost

myswitch::configuration spvx spvcc spans> new 4a1 0 32 198.29.22.46 1a2 0 100
```

This creates a SPANS SPVC where the ATM portion of the interworking connection is supported via the SPVC logic.

D.5.4.2 Creating a Frame Relay PNNI SPVC

To create a Frame Relay PNNI SPVC, perform the following steps:

1. Configure the ingress (Frame Relay to ATM) Frame Relay PVC without specifying the **-oport**, **-ovpi**, and **-ovci** parameters. For example:

```
myswitch::configuration fratm pvc> new 4A1:00 100 -epdppd 1 -name spvc_a
```

2. Display the Frame Relay PVC so you can see what `Input Port`, `Input VPI`, and `Input VCI` values were assigned to the PVC. For example:

```
myswitch::configuration fratm pvc> show
             Input          Output
 SvcId  dlci Port  VPI VCI  Port VPI VCI Admin  Epd Frr Frf8 IngRE     Name
 4A1:00  100   4A1   0  32                  up   1   0    0   disable  spvc_a
```

3. Configure a new PNNI SPVC where the *<port>*, *<vpi>*, and *<vci>* values are the `Input Port`, `Input VPI`, and `Input VCI` values that are displayed in step 2. The `dest` parameters identify the endpoint of the connection. For example:

```
myswitch::configuration spvx spvcc pnni> new 4a1 0 32
47.0005.80.ffe100.0000.f21b.19cd:1d1 -destvpi 0 -destvci 100 -name spvc_a
```

NOTE

Usually, when creating a PNNI SPVC, the **-destvpi** and **-destvci** parameters are optional (i.e., if these values are not supplied the terminating switch uses the same VPI and VCI values as the originating switch). However, when at least one endpoint of the PNNI SPVC is a port on a *FramePlus* network module, you must specify these parameters.

This creates a bi-directional PNNI SPVC from the source to the destination where the ATM portion of the interworking connection is supported via the SPVC logic.

D.6 Configuring FUNI

To configure FUNI on a *FramePlus* network module, these steps must be performed in the following order:

1. Change the application key, if necessary.
2. Choose or create any profiles that you wish to use.
3. Create each service that you need and associate the profile(s) with it.
4. Create each PVC and associate the profile(s) and service(s) with it.

These steps are described in detail in the following sections. There are also steps described for configuring PNNI SPVCs on a *FramePlus* network module.

D.6.1 Changing the Application Key

This command lets you configure a *FramePlus* network module to run either Frame Relay over ATM or to run frame-based User to Network Interface (FUNI). The network module can only run one application or the other at a time. However, you can have some network modules running FUNI and some running Frame Relay in the same switch fabric.

FramePlus network modules run Frame Relay by default. To run FUNI services on the network module, please contact FORE’s Technical Assistance Center for a valid FUNI key. To change from FUNI back to Frame Relay, the key is **`fratm170358`**. The `Appln` field under **`conf module show`** displays which application is currently running.

```
myswitch::configuration module fram> show
Module Appln Appl  Boot  Stats    Oam      Operational      Product
             swRel swRel monitor  monitor  state            number
4A     fratm 1.3.0 1.3.0 enabled  enabled  appluprunning    NMFR-4/DS1A
```

The application key should be specified only if you want to reconfigure the network module to run a different type of application. When you change the application, the switch deletes all existing services and PVCs that use a different application, and removes them from the CDB (i.e., if you are changing from Frame Relay to FUNI, the switch deletes existing Frame Relay information, and vice versa). To change the application, first admin the module down:

```
myswitch::configuration module> admin 2B down
Disabling the network module will destroy all existing connections
going through it.
Disable the network module [n]? y

myswitch::configuration module> fram app 4a XXXXXXXXXX
Changing application may cause deletion of service/connection on switch and CDB.
Proceed [n]? y
```

D.6.2 Creating the Profiles for FUNI

First, you need to decide which profiles you want to use. A default profile is provided for your convenience. If a service or connection is created without specifying any profile, the default profile in the list is used.

To display the default profile, use the **conf profile funi show** command. If this default does not match your network's needs, you can create your own using the **new** command under each profile; e.g., **conf profile funi new**. The values that you choose for [**-minVci** *<vci>*] and [**-maxVci** *<vci>*] determine the range of VCIs that you can use for *<fvci>* under **conf funi pvc new**. (See Part 2 of the *AMI Configuration Commands Reference Manual* for more information about the parameters.) Once you have determined which profile(s) to use, you can move on to Section D.6.3.

D.6.3 Creating FUNI Services

Create any FUNI services that you want and apply the FUNI and service profiles. For example:

```
myswitch::configuration funi> new <port> <timeslots> [-funi <index>]
[-service <index>] [-admin (up|down)] [-name <name>] [-id <bnp:cc>]

myswitch::configuration funi> new 4a1 1 -funi 1 -name service_a
The newly created service id is 4A1:00

myswitch::configuration funi> new 4a1 2 -service 1 -name service_b
The newly created service id is 4A1:01

myswitch::configuration funi> new 4a1 3 -name service_c
The newly created service id is 4A1:02

myswitch::configuration funi> show
Searching for FUNI services
 SvcId  TimeSlot   Admin Funi Serv Signal   Traps    Stats    Name
 4A1:00 1          up       1    0 nonexist enabled  disabled service_a
 4A1:01 2          up       0    1 nonexist enabled  disabled service_b
 4A1:02 3          up       0    0 nonexist enabled  disabled service_c
```

As you can see in the `Stats` field, the collection of statistics is disabled, by default, at the service level. It is enabled, by default, at the module level. If you want to collect statistics, you must enable that functionality here at the service level as well:

```
myswitch::configuration funi> stats <serviceid> (enable|disable)
myswitch::configuration funi> stats 4A1:00 enable
myswitch::configuration funi> stats 4A1:01 enable
myswitch::configuration funi> stats 4A1:02 enable

myswitch::configuration module fram> stats <module> (enabled|disabled)

myswitch::configuration module fram> stats 4a enabled
myswitch::configuration funi> show
Searching for FUNI services
 SvcId  TimeSlot   Admin Funi Serv Signal   Traps    Stats    Name
 4A1:00 1          up       1    0 nonexist enabled  enabled  service_a
 4A1:01 2          up       0    1 nonexist enabled  enabled  service_b
 4A1:02 3          up       0    0 nonexist enabled  enabled  service_c
```

D.6.4 Creating FUNI PVCs

Now that the profiles and services have been created, you can create your FUNI PVCs and apply the EPD / PPD profiles to the PVCs. For example:

```
myswitch::configuration funi pvc> new <serviceid> <fvpi> <fvci> [-oport <oport>]
[-ovpi <ovpi>] [-ovci <ovci>] [-faupc <index>] [-afupc <index>] [-epdppd <index>]
[-admin (up|down)] [-name <name>]

note: if oport/ovpi/ovci aren't specified, a dangling FRATM PVC will be created.
```

NOTE

The default minimum VCI is 32 and the default maximum VCI is 63. If you are trying to create a PVC on a FUNI service that uses the default FUNI service profile, you are limited to this VCI range when specifying the `<fvci>` parameter.

NOTE

Currently, the *FramePlus* network module does not permit the establishment of Frame-to-Frame connections across a single fabric. This restriction prevents connections between FramePlus ports located on the same network module, or *FramePlus* ports located across separate modules, but attached to the same switch fabric.

```
myswitch::configuration funi pvc> new 4A1:00 0 40 -oport 4c1 -ovpi 0 -ovci 40
-name pvc_a

myswitch::configuration funi pvc> new 4A1:01 0 41 -oport 4c1 -ovpi 0 -ovci 41
-name pvc_b

myswitch::configuration funi pvc> new 4A1:02 0 42 -oport 4c1 -ovpi 0 -ovci 42
-name pvc_c

myswitch::configuration funi pvc> show
        FUNI FUNI Input        Output
 SvcId   VPI  VCI  Port VPI VCI  Port VPI VCI  Admin Eppd Name
 4A1:02    0   42   4A1  32  42   4C1   0  42     up    0 pvc_c
 4A1:02    0   42   4C1   0  42   4A1  32  42     up    0 pvc_c
 4A1:00    0   40   4A1   0  40   4C1   0  40     up    0 pvc_a
 4A1:00    0   40   4C1   0  40   4A1   0  40     up    0 pvc_a
 4A1:01    0   41   4A1  16  41   4C1   0  41     up    0 pvc_b
 4A1:01    0   41   4C1   0  41   4A1  16  41     up    0 pvc_b
```

D.6.5 Configuring FUNI SPVCs

You can also configure SPVCs (Smart Permanent Virtual Circuits) on a *FramePlus* network module. An SPVC is a connection that spans multiple switch fabrics and looks like a PVC at the local and remote endpoints with an SVC in the middle. If a link carrying an SPVC goes down and there is an alternate route, then the end switch fabrics of the SPVC automatically reroute the SPVC around the failed link.

However, the procedure for creating both SPANS SPVCs and PNNI SPVCs on a *FramePlus* network module is slightly different than it is for other network modules. Section D.6.5.1 describes how to create a FUNI SPANS SPVC. Section D.6.5.2 describes how to create a FUNI PNNI SPVC.

D.6.5.1 Creating a FUNI SPANS SPVC

To create a FUNI SPANS SPVC, perform the following steps:

1. Configure the ingress (Frame Relay to ATM) FUNI PVC without specifying the **`-oport`**, **`-ovpi`**, and **`-ovci`** parameters. For example:

```
myswitch::configuration funi pvc> new 4A1:00 0 40 -epdppd 1 -name spvc_a
```

2. Display the FUNI PVC so you can see what `Input Port`, `Input VPI`, and `Input VCI` values were assigned to the PVC. For example:

```
myswitch::configuration funi pvc> show
        FUNI FUNI Input        Output
 SvcId   VPI  VCI  Port VPI VCI  Port VPI VCI  Admin Eppd Name
 4A1:00    0   40   4A1   0  40                   up    0 pvc_a
```

3. Configure a new SPANS SPVC where the *`<port>`*, *`<vpi>`*, and *`<vci>`* values are the `Input Port`, `Input VPI`, and `Input VCI` values that are displayed in step 2. The `dest` parameters identify the endpoint of the connection. For example:

```
myswitch::> open 198.29.22.46 private
Opening a session for “198.29.22.46”, please wait...
Connected to “198.29.22.46” (asx200bx).

*fishtank::> localhost

myswitch::configuration spvx spvcc spans> new 4a1 0 40 198.29.22.46 1a2 0 100
```

This creates a SPANS SPVC where the ATM portion of the interworking connection is supported via the SPVC logic.

D.6.5.2 Creating a FUNI PNNI SPVC

To create a FUNI PNNI SPVC, perform the following steps:

1. Configure the ingress (Frame Relay to ATM) FUNI PVC without specifying the **-oport**, **-ovpi**, and **-ovci** parameters. For example:

```
myswitch::configuration funi pvc> new 4A1:00 0 40 -epdppd 1 -name spvc_a
```

2. Display the FUNI PVC so you can see what `iport`, `ivpi`, and `ivci` values were assigned to the PVC. For example:

```
myswitch::configuration funi pvc> show
        FUNI FUNI Input        Output
 SvcId   VPI  VCI  Port VPI VCI  Port VPI VCI  Admin Eppd Name
 4A1:00    0   40   4A1   0  40                   up    0 pvc_a
```

3. Configure a new PNNI SPVC where the *<port>*, *<vpi>*, and *<vci>* values are the `iport`, `ivpi`, and `ivci` values that are displayed in step 2. The `dest` parameters identify the endpoint of the connection. For example:

```
myswitch::configuration spvx spvcc pnni> new 4c1 0 100
47.0005.80.ffe100.0000.f21b.19cd:4a1 -destvpi 0 -destvci 40 -name spvc_a
```

NOTE

Usually, when creating a PNNI SPVC, the **-destvpi** and **-destvci** parameters are optional (i.e., if these values are not supplied the terminating switch uses the same VPI and VCI values as the originating switch). However, when at least one endpoint of the PNNI SPVC is a port on a *FramePlus* network module, you must specify these parameters.

This creates a bi-directional PNNI SPVC from the source to the destination where the ATM portion of the interworking connection is supported via the SPVC logic.

D.7 Upgrading the *FramePlus* Network Module Software

Because the *FramePlus* network module has an on-board i960 processor, you can upgrade the the network module application software using TFTP.

NOTE

If you upgrade the switch software to *ForeThought* 6.0, all of the *FramePlus* network modules must also be upgraded to *FramePlus* software application version 1.3.0 using **`conf mod fram upgrade`**. If the *FramePlus* network modules and the switch SCP(s) are not running corresponding releases, the protocols are incompatible and an error message results.

The method for upgrading the network module software is similar to that for upgrading the switch software.

```
myswitch::configuration module fram> upgrade <module> <remotehost>:<fullpath to
remotefile>
```

These parameters are defined as follows:

Parameter	Description
module	The *FramePlus* network module on which you want to upgrade the software.
remotehost	The IP address of the remote host on which the upgrade file resides.
full path to remotefile	The full path name of the upgrade file.

For example:

```
myswitch::configuration module fram> upgrade 1A 124.11.1.13:/net/mercFCS01

WARNING: Do not remove the netmod while upgrading

{File download to netmod 1A successful}
Transfer successful.

Reset the network module [n]? y
```

To display the current revision number of the application software, use the following command:

```
myswitch::configuration module> show
Module Appln Appl  Boot  Stats    Oam      Operational      Product
             swRel swRel monitor  monitor  state            number
4D     fratm 1.3.0 1.3.0 enabled  enabled  appluprunning    NMFR-4/DS1A
```

Since this command uses TFTP as the transfer protocol, the remote host on which the upgrade file resides must be a tftpboot server. If you are unsure of how to configure the bootp server and the tftpboot server properly, see Chapter 4 of the *ATM Switch Installation and Maintenance Manual*.

Acronyms

The networking terms in the following list are defined in the Glossary of this manual. Glossary items are listed alphabetically according to the full term.

AAL	ATM Adaptation Layer
ABR	Available Bit Rate
ACM	Address Complete Message
ACR	Allowable Cell Rate
ADPCM	Adaptive Differential Pulse Code Modulation
AHFG	ATM-attached Host Functional Group
AIMUX	ATM Inverse Multiplexing
AIS	Alarm Indication Signal
AMI	Alternate Mark Inversion
AMI	ATM Management Interface
ANSI	American National Standards Institute
APCM	Adaptive Pulse Code Modulation
API	Application Program Interface
APP	Application Program
APS	Automatic Protection Switching
ARP	Address Resolution Protocol
ASCII	American Standard Code for Information Interchange
ATDM	Asynchronous Time Division Multiplexing
ATM	Asynchronous Transfer Mode
AUI	Attachment User Interface
B8ZS	Bipolar 8 Zero Substitution
BCOB	Broadband Connection Oriented Bearer
BCOB-A	Bearer Class A
BCOB-C	Bearer Class C
BCOB-X	Bearer Class X
BECN	Backward Explicit Congestion Notification
BER	Bit Error Rate
BES	Bursty Errored Seconds
BGP	Border Gateway Protocol
B-ICI	B-ISDN Inter-Carrier Interface.
BIP	Bit Interleaved Parity
B-ISDN	Broadband Integrated Services Digital Network
B-ISUP	Broadband ISDN User’s Part

BITS	Building Integrated Timing Supply
BNC	Bayonet-Neill-Concelman
BPDU	Bridge Protocol Data Unit
bps	Bits per Second
BPV	Bipolar Violation
B-TE	Broadband Terminal Equipment
BUS	Broadcast and Unknown Server
CAC	Connection Admission Control
CAS	Channel Associated Signaling
CBDS	Connectionless Broadband Data Service
CBR	Constant Bit Rate
CCITT	International Telephone and Telegraph Consultative Committee
CCS	Common Channel Signaling
CDV	Cell Delay Variation
CE	Connection Endpoint
CEI	Connection Endpoint Identifier
CES	Circuit Emulation Service
CGA	Carrier Group Alarm
CIP	Carrier Identification Parameter
CIR	Committed Information Rate
CLIP	Classical IP
CLP	Cell Loss Priority
CLR	Cell Loss Ratio-1-15
CLS	Connectionless service
CMIP	Common Management Interface Protocol
CMR	Cell Misinsertion Rate
CPE	Customer Premise Equipment
CRA	Cell Rate Adaptation
CRC	Cyclic Redundancy Check
CRS	Cell Relay Service
CS	Controlled Slip, **or** Convergence Sublayer
CSU	Channel Service Unit
CTD	Cell Transfer Delay
CTS	Clear To Send
DACS	Digital Access and Cross-Connect System
DARPA	Defense Advanced Research Projects Agency
DCC	Data Country Code
DCE	Data Communications Equipment
DCS	Digital Cross-connect System
DES	Destination End Station
DFA	DXI Frame Address
DLCI	Data Link Connection Identifier

DNS	Domain Naming System
DSn	Digital Standard n (n=0, 1, 1C, 2, and 3)
DSR	Data Set Ready
DTE	Data Terminal Equipment
DTR	Data Terminal Ready
EEPROM	Electrically Erasable Programmable Read Only Memory
EFCI	Explicit Forward Congestion Indication
EGP	Exterior Gateway Protocol
EIA	Electronics Industries Association
EISA	Extended Industry Standard Architecture
ELAN	Emulated Local Area Network
EMI	Electromagnetic Interference
EPROM	Erasable Programmable Read Only Memory
EQL	Equalization
ER	Explicit Rate
ES	End System, **or** Errored Second
ESF	Extended Super Frame
ESI	End System Identifier
EXZ	Excessive Zeroes (Error Event)
FC	Face Contact
FCC	Federal Communications Commission
FCS	Frame Check Sequence
FDDI	Fiber Distributed Data Interface
FDM	Frequency Division Multiplexing
FEBE	Far End Block Error
FEC	Forward Error Correction
FECN	Forward Explicit Congestion Notification
FERF	Far End Receive Failure
FIFO	First-In, First-Out
FRS	Frame-Relay Service
FTP	File Transfer Protocol
FT-PNNI	*ForeThought* PNNI
FUNI	Frame-Based UNI
GCAC	Generic Connection Admission Control
GCRA	Generic Cell Rate Algorithm
GFC	Generic Flow Control
HDB3	High Density Bipolar
HDLC	High Level Data Link Control
HEC	Header Error Control
HIPPI	High Performance Parallel Interface
HSSI	High-Speed Serial Interface
ICMP	Internet Control Message Protocol

IDU	Interface Data Unit
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ILMI	Interim Local Management Interface
IP	Internet Protocol
IPX	Internetwork Packet Exchange
IS	Intermediate system
ISDN	Integrated Services Digital Network
ISO	International Standards Organization
ITU-T	International Telecommunication Union Telecommunication
IWF	Interworking Function
IXC	Interexchange Carriers
JPEG	Joint Photographic Experts Group
Kbps	Kilobits per second
LAN	Local Area Network
LANE	LAN Emulation
LAPB	Link Access Procedure, Balanced
LATA	Local Access and Transport Area
LBO	Line Build Out
LCV	Line Code Violations
LE_ARP	LAN Emulation Address Resolution Protocol
LEC	LAN Emulation Client
LECS	LAN Emulation Configuration Server
LES	LAN Emulation Server
LLC	Logical Link Control
LOF	Loss Of Frame
LOP	Loss Of Pointer
LOS	Loss Of Signal
LSB	Least Significant Bit
MAC	Media Access Control
MAN	Metropolitan Area Network
MAU	Media Attachment Unit
MBS	Maximum Burst Size
MCDV	Maximum Cell Delay Variance
MCLR	Maximum Cell Loss Ratio
MCR	Minimum Cell Rate
MCTD	Maximum Cell Transfer Delay
MIB	Management Information Base
MIC	Media Interface Connector
MID	Message Identifier
MMF	Multimode Fiber Optic Cable
MPEG	Motion Picture Experts Group
MPOA	Multiprotocol over ATM

MSB	Most Significant Bit
MTU	Maximum Transmission Unit
NM	Network Management Entity
NML	Network Management Layer
NMS	Network Management Station
NNI	Network-to-Network Interface or Network Node Interface
NPC	Network Parameter Control
NRZ	Non Return to Zero
NRZI	Non Return to Zero Inverted
NSAP	Network Service Access Point
NTSC	National TV Standards Committee
OAM	Operation and Maintenance Cell
OC-n	Optical Carrier level-n
OID	Object Identifier
OOF	Out-of-Frame
OSI	Open Systems Interconnection
OSPF	Open Shortest Path First Protocol
OUI	Organizationally Unique Identifier
PAD	Packet Assembler Disassembler
PAL	Phase Alternate Line
PBX	Private Branch Exchange
PCI	Peripheral Component Interconnect
PCM	Pulse Code Modulation
PCR	Peak Cell Rate
PDN	Public Data Network
PDU	Protocol Data Unit
PHY	Physical Layer
ping	Packet Internet Groper
PLCP	Physical Layer Convergence Protocol
PLP	Packet Level Protocol
PM	Physical Medium
PMD	Physical Medium Dependent
PNNI	Private Network Node Interface or Private Network-to-Network Interface
PPP	Point-to-Point Protocol
PROM	Programmable Read-Only Memory
PRS	Primary Reference Source
PSN	Packet Switched Network
PT	Payload Type
PVC	Permanent Virtual Circuit (or Channel)
PVCC	Permanent Virtual Channel Connection
PVPC	Permanent Virtual Path Connection
QD	Queuing Delay
QoS	Quality of Service

RD	Routing Domain
RFCs	Requests For Comment
RFI	Radio Frequency Interference
RIP	Routing Information Protocol
RISC	Reduced Instruction Set Computer
RTS	Request To Send
SA	Source Address
SA	Source MAC Address
SAP	Service Access Point
SAR	Segmentation And Reassembly
SC	Structured Cabling, **or** Structured Connectors, **or** Stick and Click
SCR	Sustainable Cell Rate
SCSI	Small Computer Systems Interface
SDLC	Synchronous Data Link Control
SDU	Service Data Unit
SEAL	Simple and Efficient Adaptation Layer
SECAM	Systeme En Coleur Avec Memoire
SEL	Selector
SES	Severely Errored Seconds
SF	Super Frame
SGMP	Simple Gateway Management Protocol
SIR	Sustained Information Rate
SLIP	Serial Line IP
SMDS	Switched Multimegabit Data Service
SMF	Single Mode Fiber
SMTP	Simple Mail Transfer Protocol
SNA	Systems Network Architecture
SNAP	SubNetwork Access Protocol
SNI	Subscriber Network Interface
SNMP	Simple Network Management Protocol
SONET	Synchronous Optical Network
SPANS	Simple Protocol for ATM Network Signalling
SPARC	Scalable Processor Architecture Reduced instruction set Computer
SPE	Synchronous Payload Envelope
SPVC	Smart PVC
SS7	Signaling System No. 7
SSCOP	Service Specific Connection Oriented Protocol
SSCS	Service Specific Convergence Sublayer
ST	Straight Tip, **or** Stick and Turn
STM	Synchronous Transfer Mode

STP	Shielded Twisted Pair, Spanning Tree Protocol
STS	Synchronous Transport Signal
SVC	Switched Virtual Circuit (or Channel)
SVCC	Switched Virtual Channel Connection
SVPC	Switched Virtual Path Connection
TAXI	Transparent Asynchronous Transmitter / Receiver Interface
TC	Transmission Convergence
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
TCR	Tagged Cell Rate
TCS	Transmission Convergence Sublayer
TDM	Time Division Multiplexing
TE	Terminal Equipment
TFTP	Trivial File Transfer Protocol
TM	Traffic Management
UAS	Unavailable Seconds
UBR	Unspecified Bit Rate
UDP	User Datagram Protocol
UNI	User-to-Network Interface
UPC	Usage Parameter Control
UTOPIA	Universal Test & Operations Interface for ATM
UTP	Unshielded Twisted Pair
VBR	Variable Bit Rate
VC	Virtual Channel (or Circuit)
VCC	Virtual Channel Connection
VCI	Virtual Channel Identifier
VCL	Virtual Channel Link
VINES	Virtual Network Software
VLAN	Virtual Local Area Network
VP	Virtual Path
VPC	Virtual Path Connection
VPDN	Virtual Private Data Network
VPI	Virtual Path Identifier
VPL	Virtual Path Link
VPN	Virtual Private Network
VPT	Virtual Path Terminator
VS/VD	Virtual Source / Virtual Destination
VT	Virtual Tributary
WAN	Wide-Area Network
ZBTSI	Zero Byte Time Slot Interchange

Glossary

10Base-T - a 10 Mbps baseband Ethernet specification utilizing twisted-pair cabling (Category 3, 4, or 5). 10BaseT, which is part of the IEEE 802.3 specification, has a distance limit of approximately 100 meters per segment.

802.1d Spanning Tree Bridging - the IEEE standard for bridging; a MAC layer standard for transparently connecting two or more LANs (often called subnetworks) that are running the same protocols and cabling. This arrangement creates an extended network, in which any two workstations on the linked LANs can share data.

802.3 Ethernet - the IEEE standard for Ethernet; a physical-layer standard that uses the CSMA / CD access method on a bus-topology LAN.

802.5 Token Ring - the IEEE physical-layer standard that uses the token-passing access method on a ring-topology LAN.

AAL Connection - an association established by the AAL between two or more next higher layer entities.

Adapter - A fitting that supplies a passage between two sets of equipment when they cannot be directly interconnected.

Adaptive Differential Pulse Code Modulation (ADPCM) - A technique that allows analog voice signals to be carried on a 32K bps digital channel. Sampling is done at 8Hz with 4 bits used to describe the difference between adjacent samples.

Adaptive Pulse Code Modulation (APCM) - A technique that effectively reduces occupied bandwidth per active speaker by reducing sampling rates during periods of overflow peak traffic.

Address - A unique identity of each network station on a LAN or WAN.

Address Complete Message (ACM) - A B-ISUP call control message from the receiving exchange to sending exchange indicating the completion of address information.

Address Mask - a bit mask used to identify which bits in an address (usually an IP address) are network significant, subnet significant, and host significant portions of the complete address. This mask is also known as the subnet mask because the subnetwork portion of the address can be determined by comparing the binary version of the mask to an IP address in that subnet. The mask holds the same number of bits as the protocol address it references.

Address Prefix - A string of 0 or more bits up to a maximum of 152 bits that is the lead portion of one or more ATM addresses.

Address Resolution - The procedure by which a client associates a LAN destination with the ATM address of another client or the BUS.

Address Resolution Protocol (ARP) - a method used to resolve higher level protocol addressing (such as IP) into the appropriate header data required for ATM; i.e., port, VPI, and VCI; also defines the AAL type to be used.

Agent - a component of network- and desktop-management software, such as SNMP, that gathers information from MIBs.

alarm - an unsolicited message from a device, typically indicating a problem with the system that requires attention.

Alarm Indication Signal (AIS) - In T1, an all ones condition used to alert a receiver that its incoming signal (or frame) has been lost. The loss of signal or frame is detected at the receiving end, and the failed signal is replaced by all the ones condition which the receiver interprets as an AIS. The normal response to this is AIS is for the receiving end to generate a yellow alarm signal as part of its transmission towards the faulty end. (The AIS itself is sometimes called a Blue Signal).

A-Law - The PCM coding and companding standard used in Europe.

Allowable Cell Rate (ACR) - parameter defined by the ATM Forum for ATM traffic management. ACR varies between the MCR and the PCR, and is dynamically controlled using congestion control mechanisms.

Alternate Mark Inversion (AMI) - A line coding format used on T1 facilities that transmits ones by alternate positive and negative pulses.

Alternate Routing - A mechanism that supports the use of a new path after an attempt to set up a connection along a previously selected path fails.

American National Standards Institute (ANSI) - a private organization that coordinates the setting and approval of some U.S. standards. It also represents the United States to the International Standards Organization.

American Standard Code for Information Interchange (ASCII) - a standard character set that (typically) assigns a 7-bit sequence to each letter, number, and selected control characters.

AppleTalk - a networking protocol developed by Apple Computer for communication between Apple's products and other computers. Independent of the network layer, AppleTalk runs on LocalTalk, EtherTalk and TokenTalk.

Application Layer - Layer seven of the ISO reference model; provides the end-user interface.

Application Program (APP) - a complete, self-contained program that performs a specific function directly for the user.

Application Program Interface (API) - a language format that defines how a program can be made to interact with another program, service, or other software; it allows users to develop custom interfaces with FORE products.

Assigned Cell - a cell that provides a service to an upper layer entity or ATM Layer Management entity (ATMM-entity).

asxmon - a FORE program that repeatedly displays the state of the switch and its active ports.

Asynchronous Time Division Multiplexing (ATDM) - a multiplexing technique in which a transmission capability is organized into a priori, unassigned time slots. The time slots are assigned to cells upon request of each application's instantaneous real need.

Asynchronous Transfer Mode (ATM) - a transfer mode in which the information is organized into cells. It is asynchronous in the sense that the recurrence of cells containing information from an individual user is not necessarily periodic.

ATM Adaptation Layer (AAL) - the AAL divides user information into segments suitable for packaging into a series of ATM cells. AAL layer types are used as follows:

AAL-1 - constant bit rate, time-dependent traffic such as voice and video
AAL-2 - still undefined; a placeholder for variable bit rate video transmission
AAL-3/4 - variable bit rate, delay-tolerant data traffic requiring some sequencing and/or error detection support (originally two AAL types, connection-oriented and connectionless, which have been combined)
AAL-5 - variable bit rate, delay-tolerant, connection-oriented data traffic requiring minimal sequencing or error detection support

ATM Address - Defined in the UNI Specification as 3 formats, each having 20 bytes in length.

ATM Forum - an international non-profit organization formed with the objective of accelerating the use of ATM products and services through a rapid convergence of interoperability specifications. In addition, the Forum promotes industry cooperation and awareness.

ATM Inverse Multiplexing (AIMUX) - A device that allows multiple T1 or E1 communications facilities to be combined into a single broadband facility for the transmission of ATM cells.

ATM Layer link - a section of an ATM Layer connection between two adjacent active ATM Layer entities (ATM-entities).

ATM Link - a virtual path link (VPL) or a virtual channel link (VCL).

ATM Management Interface (AMI) - the user interface to FORE Systems' *ForeThought* switch control software (SCS). AMI lets users monitor and change various operating configurations of FORE Systems switches and network module hardware and software, IP connectivity, and SNMP network management.

ATM Peer-to-Peer Connection - a virtual channel connection (VCC) or a virtual path connection (VPC) directly established, such as workstation-to-workstation. This setup is not commonly used in networks.

ATM Traffic Descriptor - a generic list of parameters that can be used to capture the intrinsic traffic characteristics of a requested ATM connection.

ATM User-to-User Connection - an association established by the ATM Layer to support communication between two or more ATM service users (i.e., between two or more next higher layer entities or between two or more ATM entities). The communication over an ATM Layer connection may be either bidirectional or unidirectional. The same Virtual Channel Identifier (VCI) is used for both directions of a connection at an interface.

atmarp - a FORE program that shows and manipulates ATM ARP entries maintained by the given device driver. This is also used to establish PVC connections.

ATM-attached Host Functional Group (AHFG) - The group of functions performed by an ATM-attached host that is participating in the MPOA service.

atmconfig - a FORE program used to enable or disable SPANS signalling.

atmstat - a FORE program that shows statistics gathered about a given adapter card by the device driver. These statistics include ATM layer and ATM adaptation layer cell and error counts. This can also be used to query other hosts via SNMP.

Attachment User Interface (AUI) - IEEE 802.3 interface between a media attachment unit (MAU) and a network interface card (NIC). The term AUI can also refer to the rear panel port to which an AUI cable might attach.

Auto-logout - a feature that automatically logs out a user if there has been no user interface activity for a specified length of time.

Automatic Protection Switching (APS) - Equipment installed in communications systems to detect circuit failures and automatically switch to redundant, standby equipment.

Available Bit Rate (ABR) - a type of traffic for which the ATM network attempts to meet that traffic's bandwidth requirements. It does not guarantee a specific amount of bandwidth and the end station must retransmit any information that did not reach the far end.

Backbone - the main connectivity device of a distributed system. All systems that have connectivity to the backbone connect to each other, but systems can set up private arrangements with each other to bypass the backbone to improve cost, performance, or security.

Backplane - High-speed communications line to which individual components are connected.

Backward Explicit Congestion Notification (BECN) - A Resource Management cell type generated by the network or the destination, indicating congestion or approaching congestion for traffic flowing in the direction opposite that of the BECN cell.

Bandwidth - usually identifies the capacity or amount of data that can be sent through a given circuit; may be user-specified in a PVC.

Baud - unit of signalling speed, equal to the number of discrete conditions or signal events per second. If each signal event represents only one bit, the baud rate is the same as bps; if each signal event represents more than one bit (such as a dibit), the baud rate is smaller than bps.

Bayonet-Neill-Concelman (BNC) - a bayonet-locking connector used to terminate coaxial cables. BNC is also referred to as Bayonet Network Connector.

Bipolar 8 Zero Substitution (B8ZS) - a technique used to satisfy the ones density requirements of digital T-carrier facilities in the public network while allowing 64 Kbps clear channel data. Strings of eight consecutive zeroes are replaced by an eight-bit code representing two intentional bipolar pulse code violations (000V10V1).

Bipolar Violation (BPV) - an error event on a line in which the normal pattern of alternating high (one) and low (zero) signals is disrupted. A bipolar violation is noted when two high signals occur without an intervening low signal, or vice versa.

B-ISDN Inter-Carrier Interface (B-ICI) - An ATM Forum defined specification for the interface between public ATM networks to support user services across multiple public carriers.

Bit Error Rate (BER) - A measure of transmission quality, generally shown as a negative exponent, (e.g., 10^{-7} which means 1 out of 10^{7} bits [1 out of 10,000,000 bits] are in error).

Bit Interleaved Parity (BIP) - an error-detection technique in which character bit patterns are forced into parity, so that the total number of one bits is always odd or always even. This is accomplished by the addition of a one or zero bit to each byte, as the byte is transmitted; at the other end of the transmission, the receiving device verifies the parity (odd or even) and the accuracy of the transmission.

Bit Robbing - The use of the least significant bit per channel in every sixth frame for signaling.

Bit Stuffing - A process in bit-oriented protocols where a zero is inserted into a string of ones by the sender to prevent the receiver from interpreting valid user data (the string of ones) as control characters (a Flag character for instance).

Border Gateway Protocol (BGP) - used by gateways in an internet connecting autonomous networks. It is derived from experiences learned using the EGP.

bps - bits per second

Bridge - a device that expands a Local Area Network by forwarding frames between data link layers associated with two separate cables, usually carrying a common protocol. Bridges can usually be made to filter certain packets (to forward only certain traffic).

Bridge Protocol Data Unit (BPDU) - A message type used by bridges to exchange management and control information.

Broadband - a service or system requiring transmission channels capable of supporting rates greater than the Integrated Services Digital Network (ISDN) primary rate.

Broadband Access - an ISDN access capable of supporting one or more broadband services.

Broadband Connection Oriented Bearer (BCOB) - Information in the SETUP message that indicates the type of service requested by the calling user.

BCOB-A (Bearer Class A) - Indicated by ATM end user in SETUP message for connection-oriented, constant bit rate service. The network may perform internetworking based on AAL information element (IE).

BCOB-C (Bearer Class C) - Indicated by ATM end user in SETUP message for connection-oriented, variable bit rate service. The network may perform internetworking based on AAL information element (IE).

BCOB-X (Bearer Class X) - Indicated by ATM end user in SETUP message for ATM transport service where AAL, traffic type and timing requirements are transparent to the network.

Broadband Integrated Services Digital Network (B-ISDN) - a common digital network suitable for voice, video, and high-speed data services running at rates beginning at 155 Mbps.

Broadband ISDN User's Part (B-ISUP) - A protocol used to establish, maintain and release broadband switched network connections across an SS7/ATM network.

Broadband Terminal Equipment (B-TE) - An equipment category for B-ISDN which includes terminal adapters and terminals.

Broadcast - Data transmission to all addresses or functions.

Broadcast and Unknown Server (BUS) - in an emulated LAN, the BUS is responsible for accepting broadcast, multicast, and unknown unicast packets from the LECs to the broadcast MAC address (FFFFFFFFFFFF) via dedicated point-to-point connections, and forwarding the packets to all of the members of the ELAN using a single point-to-multipoint connection.

Brouter (bridging/router) - a device that routes some protocols and bridges others based on configuration information.

Buffer - A data storage medium used to compensate of a difference in rate of data flow or time of occurrence of events when transmitting data from one device to another.

Building Integrated Timing Supply (BITS) - a master timing supply for an entire building, which is a master clock and its ancillary equipment. The BITS supplies DS1 and/or composite clock timing references for synchronization to all other clocks and timing sources in that building.

Bursty Errored Seconds (BES) - a BES contains more than 1 and fewer than 320 path coding violation error events, and no severely errored frame or AIS defects. Controlled slips are not included in determining BESs.

Bursty Second - a second during which there were at least the set number of BES threshold event errors but fewer than the set number of SES threshold event errors.

Byte - A computer-readable group of bits (normally 8 bits in length).

Call - an association between two or more users or between a user and a network entity that is established by the use of network capabilities. This association may have zero or more connections.

Carrier - a company, such as any of the "baby Bell" companies, that provide network communications services, either within a local area or between local areas.

Carrier Group Alarm (CGA) - A service alarm generated by a channel bank when an out-of-frame (OOF) condition exists for some predetermined length of time (generally 300 milliseconds to 2.5 seconds). The alarm causes the calls using a trunk to be dropped and trunk conditioning to be applied.

Carrier Identification Parameter (CIP) - A 3 or 4 digit code in the initial address message identifying the carrier to be used for the connection.

cchan - a FORE program that manages virtual channels on a *ForeRunner* switch running asxd.

Cell - an ATM Layer protocol data unit (PDU). The basic unit of information transported in ATM technology, each 53-byte cell contains a 5-byte header and a 48-byte payload.

Cell Delay Variation (CDV) - a quantification of cell clumping for a connection. The cell clumping CDV (yk) is defined as the difference between a cell's expected reference arrival time (ck) and its actual arrival time (ak). The expected reference arrival time (ck) of cell k of a specific connection is max. T is the reciprocal of the negotiated peak cell rate.

Cell Delineation - the protocol for recognizing the beginning and end of ATM cells within the raw serial bit stream.

Cell Header - ATM Layer protocol control information.

Cell Loss Priority (CLP) - the last bit of byte four in an ATM cell header; indicates the eligibility of the cell for discard by the network under congested conditions. If the bit is set to 1, the cell may be discarded by the network depending on traffic conditions.

Cell Loss Ratio - In a network, cell loss ratio is (1-x/y), where y is the number of cells that arrive in an interval at an ingress of the network; and x is the number of these y cells that leave at the egress of the network element.

Cell Loss Ratio (CLR) - CLR is a negotiated QoS parameter and acceptable values are network specific. The objective is to minimize CLR provided the end-system adapts the traffic to the changing ATM layer transfer characteristics. The Cell Loss Ratio is defined for a connection as: Lost Cells/Total Transmitted Cells. The CLR parameter is the value of CLR that the network agrees to offer as an objective over the lifetime of the connection. It is expressed as an order of magnitude, having a range of 10-1 to 10-15 and unspecified.

Cell Misinsertion Rate (CMR) - the ratio of cells received at an endpoint that were not originally transmitted by the source end in relation to the total number of cells properly transmitted.

Cell Rate Adaptation (CRA) - a function performed by a protocol module in which empty cells (known as unassigned cells) are added to the output stream. This is because there always must be a fixed number of cells in the output direction; when there are not enough cells to transmit, unassigned cells are added to the output data stream.

Cell Relay Service (CRS) - a carrier service which supports the receipt and transmission of ATM cells between end users in compliance with ATM standards and implementation specifications.

Cell Transfer Delay - the transit delay of an ATM cell successfully passed between two designated boundaries. See CTD.

Cell Transfer Delay (CTD) - This is defined as the elapsed time between a cell exit event at the measurement point 1 (e.g., at the source UNI) and the corresponding cell entry event at the measurement point 2 (e.g., the destination UNI) for a particular connection. The cell transfer delay between two measurement points is the sum of the total inter-ATM node transmission delay and the total ATM node processing delay.

Channel - A path or circuit along which information flows.

Channel Associated Signaling (CAS) - a form of circuit state signaling in which the circuit state is indicated by one or more bits of signaling status sent repetitively and associated with that specific circuit.

Channel Bank - A device that multiplexes many slow speed voice or data conversations onto high speed link and controls the flow.

Channel Service Unit (CSU) - An interface for digital leased lines which performs loopback testing and line conditioning.

Channelization - capability of transmitting independent signals together over a cable while still maintaining their separate identity for later separation.

Circuit - A communications link between points.

Circuit Emulation Service (CES) - The ATM Forum circuit emulation service interoperability specification specifies interoperability agreements for supporting Constant Bit Rate (CBR) traffic over ATM networks that comply with the other ATM Forum interoperability agreements. Specifically, this specification supports emulation of existing TDM circuits over ATM networks.

Classical IP (CLIP) - IP over ATM which conforms to RFC 1577.

Clear to Send (CTS) - and RS-232 modem interface control signal (sent from the modem to the DTE on pin 5) which indicates that the attached DTE may begin transmitting; issuance in response to the DTE's RTS.

Clocking - Regularly timed impulses.

Closed User Group - A subgroup of network users that can be its own entity; any member of the subgroup can only communicate with other members of that subgroup.

Coaxial Cable - Coax is a type of electrical communications medium used in the LAN environment. This cable consists of an outer conductor concentric to an inner conductor, separated from each other by insulating material, and covered by some protective outer material. This medium offers large bandwidth, supporting high data rates with high immunity to electrical interference and a low incidence of errors. Coax is subject to distance limitations and is relatively expensive and difficult to install.

Cold Start Trap - an SNMP trap which is sent after a power-cycle (see *trap*).

Collision - Overlapping transmissions that occur when two or more nodes on a LAN attempt to transmit at or about the same time.

Committed Information Rate (CIR) - CIR is the information transfer rate which a network offering Frame Relay Services (FRS) is committed to transfer under normal conditions. The rate is averaged over a minimum increment of time.

Common Channel Signaling (CCS) - A form signaling in which a group of circuits share a signaling channel. Refer to SS7.

Common Management Interface Protocol (CMIP) - An ITU-TSS standard for the message formats and procedures used to exchange management information in order to operate, administer maintain and provision a network.

Concatenation - The connection of transmission channels similar to a chain.

Concentrator - a communications device that offers the ability to concentrate many lower-speed channels into and out of one or more high-speed channels.

Configuration - The phase in which the LE Client discovers the LE Service.

Congestion Management - traffic management feature that helps ensure reasonable service for VBR connections in an ATM network, based on a priority, sustained cell rate (SCR), and peak cell rate (PCR). During times of congestion, bandwidth is reduced to the SCR, based on the priority of the connection.

Connection - the concatenation of ATM Layer links in order to provide an end-to-end information transfer capability to access points.

Connection Admission Control (CAC) - the procedure used to decide if a request for an ATM connection can be accepted based on the attributes of both the requested connection and the existing connections.

Connection Endpoint (CE) - a terminator at one end of a layer connection within a SAP.

Connection Endpoint Identifier (CEI) - an identifier of a CE that can be used to identify the connection at a SAP.

Connectionless Broadband Data Service (CBDS) - A connectionless service similar to Bellcore's SMDS defined by European Telecommunications Standards Institute (ETSI).

Connectionless Service - a type of service in which no pre-determined path or link has been established for transfer of information, supported by AAL 4.

Connectionless Service (CLS) - A service which allows the transfer of information among service subscribers without the need for end-to- end establishment procedures.

Connection-Oriented Service - a type of service in which information always traverses the same pre-established path or link between two points, supported by AAL 3.

Constant Bit Rate (CBR) - a type of traffic that requires a continuous, specific amount of bandwidth over the ATM network (e.g., digital information such as video and digitized voice).

Controlled Slip (CS) - a situation in which one frame's worth of data is either lost or replicated. A controlled slip typically occurs when the sending device and receiving device are not using the same clock.

Convergence Sublayer (CS) - a portion of the AAL. Data is passed first to the CS where it is divided into rational, fixed-length packets or PDUs (Protocol Data Units). For example, AAL 4 processes user data into blocks that are a maximum of 64 kbytes long.

Corresponding Entities - peer entities with a lower layer connection among them.

cpath - a FORE program used to manage virtual paths on a *ForeRunner* switch running asxd.

cport - a FORE program that monitors and changes the state of ports on a *ForeRunner* switch running asxd.

Cross Connection - a mapping between two channels or paths at a network device.

Customer Premise Equipment (CPE) - equipment that is on the customer side of the point of demarcation, as opposed to equipment that is on a carrier side. See also point of demarcation.

Cut Through - Establishment of a complete path for signaling and/or audio communications.

Cyclic Redundancy Check (CRC) - an error detection scheme in which a number is derived from the data that will be transmitted. By recalculating the CRC at the remote end and comparing it to the value originally transmitted, the receiving node can detect errors.

D3/D4 - Refers to compliance with AT&T TR (Technical Reference) 62411 definitions for coding, supervision, and alarm support. D3/D4 compatibility ensures support of digital PBXes, M24 services, Megacom services, and Mode 3 D3/D4 channel banks at DS-1 level.

D4 Channelization - refers to compliance with AT&T Technical Reference 62411 regarding DS1 frame layout (the sequential assignment of channels and time slot numbers within the DS1).

D4 Framed/Framing Format - in T1, a 193-bit frame format in which the 193rd bit is used for framing and signaling information (the frame/framing bit). To be considered in support of D4 Framing, a device must be able to synchronize and frame-up on the 193rd bit.

Data Communications Equipment (DCE) - a definition in the RS232C standard that describes the functions of the signals and the physical characteristics of an interface for a communication device such as a modem.

Data Country Code (DCC) - This specifies the country in which an address is registered. The codes are given in ISO 3166. The length of this field is two octets. The digits of the data country code are encoded in Binary Coded Decimal (BCD) syntax. The codes will be left justified and padded on the right with the hexadecimal value “F” to fill the two octets.

Data Link - Communications connection used to transmit data from a source to a destination.

Data Link Connection Identifier (DLCI) - connection identifier associated with frame relay packets that serves the same functions as, and translates directly to, the VPI/VCI on an ATM cell.

Data Link Layer - Layer 2 of the OSI model, responsible for encoding data and passing it to the physical medium. The IEEE divides this layer into the LLC (Logical Link Control) and MAC (Media Access Control) sublayers.

Data Set Ready (DSR) - an RS-232 modem interface control signal (sent from the modem to the DTE on pin 6) which indicates that the modem is connected to the telephone circuit. Usually a prerequisite to the DTE issuing RTS.

Data Terminal Equipment (DTE) - generally user devices, such as terminals and computers, that connect to data circuit-terminating equipment. They either generate or receive the data carried by the network.

Data Terminal Ready (DTR) - an RS232 modem interface control signal (sent from the DTE to the modem on pin 20) which indicates that the DTE is ready for data transmission and which requests that the modem be connected to the telephone circuit.

Datagram - a packet of information used in a connectionless network service that is routed to its destination using an address included in the datagram's header.

DECnet - Digital Equipment Corporation's proprietary LAN.

Defense Advanced Research Projects Agency (DARPA) - the US government agency that funded the ARPANET.

Demultiplexing - a function performed by a layer entity that identifies and separates SDUs from a single connection to more than one connection (see *multiplexing*).

Destination End Station (DES) - An ATM termination point which is the destination for ATM messages of a connection and is used as a reference point for ABR services. See SES.

Digital Access and Cross-Connect System (DACS) - Digital switching system for routing T1 lines, and DS-0 portions of lines, among multiple T1 ports.

Digital Cross-connect System (DCS) - an electronic patch panel used to route digital signals in a central office.

Digital Standard n (0, 1, 1C, 2, and 3) (DSn) - a method defining the rate and format of digital hierarchy, with asynchronous data rates defined as follows:

DS0	64kb/s	1 voice channel
DS1	1.544Mb/s	24 DS0s
DS1C	3.152 Mb/s	2 DS1s
DS2	6.312 Mb/s	4 DS1s
DS3	44.736 Mb/s	28 DS1s

Synchronous data rates (SONET) are defined as:

STS-1/OC-1	51.84 Mb/s	28 DS1s or 1 DS3
STS-3/OC-3	155.52 Mb/s	3 STS-1s byte interleaved
STS-3c/OC-3c	155.52 Mb/s	Concatenated, indivisible payload
STS-12/OC-12	622.08 Mb/s	12 STS-1s, 4 STS-3cs, or any mixture
STS-12c/OC-12c	622.08 Mb/s	Concatenated, indivisible payload
STS-48/OC-48	2488.32 Mb/s	48 STS-1s, 16 STS-3cs, or any mixture

DIP (Dual In-line Package) Switch - a device that has two parallel rows of contacts that let the user switch electrical current through a pair of those contacts to on or off. They are used to reconfigure components and peripherals.

Domain Name Server - a computer that converts names to their corresponding Internet numbers. It allows users to telnet or FTP to the name instead of the number.

Domain Naming System (DNS) - the distributed name and address mechanism used in the Internet.

Duplex - Two way communication.

DXI - a generic phrase used in the full names of several protocols, all commonly used to allow a pair of DCE and DTE devices to share the implementation of a particular WAN protocol. The protocols define the packet formats used to transport data between DCE and DTE devices.

DXI Frame Address (DFA) - a connection identifier associated with ATM DXI packets that serves the same functions as, and translates directly to, the VPI/VCI on an ATM cell.

Dynamic Allocation - A technique in which the resources assigned for program execution are determined by criteria applied at the moment of need.

E.164 - A public network addressing standard utilizing up to a maximum of 15 digits. ATM uses E.164 addressing for public network addressing.

E1 - Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 2.048 Mbps. E1 lines can be leased for private use from common carriers.

E3 - Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 34.368 Mbps. E3 lines can be leased for private use from common carriers.

Edge Device - A physical device which is capable of forwarding packets between legacy interworking interfaces (e.g., Ethernet, Token Ring, etc.) and ATM interfaces based on data-link and network layer information but which does not participate in the running of any network layer routing protocol. An Edge Device obtains forwarding descriptions using the route distribution protocol.

elarp - a FORE program that shows and manipulates MAC and ATM address mappings for LAN Emulation Clients (LECs).

elconfig - a FORE program that shows and modifies LEC configuration. Lets the user set the NSAP address of the LAN Emulation Configuration Server, display the list of Emulated LANs configured in the LECS for this host, display the list of ELANs locally configured along with the membership state of each, and locally administer ELAN membership.

Electrically Erasable Programmable Read Only Memory (EEPROM) - an EPROM that can be cleared with electrical signals rather than the traditional ultraviolet light.

Electromagnetic Interference (EMI) - signals generated and radiated by an electronic device that cause interference with radio communications, among other effects.

Electronics Industries Association (EIA) - a USA trade organization that issues its own standards and contributes to ANSI; developed RS-232. Membership includes USA manufacturers.

Embedded SNMP Agent - an SNMP agent can come in two forms: embedded or proxy. An embedded SNMP agent is integrated into the physical hardware and software of the unit.

Emulated Local Area Network (ELAN) - A logical network initiated by using the mechanisms defined by LAN Emulation. This could include ATM and legacy attached end stations.

End System (ES) - a system where an ATM connection is terminated or initiated (an originating end system initiates the connection; a terminating end system terminates the connection).

End System Identifier (ESI) - This identifier distinguishes multiple nodes at the same level in case the lower level peer group is partitioned.

End-to-End Connection - when used in reference to an ATM network, a connection that travels through an ATM network, passing through various ATM devices and with endpoints at the termination of the ATM network.

Enterprise - Terminology generally referring to customers with multiple, non-contiguous geographic locations.

Equalization (EQL) - the process of compensating for line distortions.

Erasable Programmable Read Only Memory (EPROM) - A PROM which may be erased and rewritten to perform new or different functions (normally done with a PROM burner).

Errored Second (ES) - a second during which at least one code violation occurred.

Ethernet - a 10-Mbps, coaxial standard for LANs in which all nodes connect to the cable where they contend for access.

Excessive Zeroes (EXZ) Error Event - An Excessive Zeroes error event for an AMI-coded signal is the occurrence of more than fifteen contiguous zeroes. For a B8ZS coded signal, the defect occurs when more than seven contiguous zeroes are detected.

Explicit Forward Congestion Indication (EFCI) - the second bit of the payload type field in the header of an ATM cell, the EFCI bit indicates network congestion to receiving hosts. On a congested switch, the EFCI bit is set to “1” by the transmitting network module when a certain number of cells have accumulated in the network module's shared memory buffer. When a cell is received that has its EFCI bit set to “1,” the receiving host notifies the sending host, which should then reduce its transmission rate.

Explicit Rate (ER) - The Explicit Rate is an RM-cell field used to limit the source ACR to a specific value. It is initially set by the source to a requested rate (such as PCR). It may be subsequently reduced by any network element in the path to a value that the element can sustain. ER is formatted as a rate.

Extended Industry Standard Architecture (EISA) - bus architecture for desktop computers that provides a 32-bit data passage and maintains compatibility with the ISA or AT architecture.

Extended Super Frame (ESF) - a T1 framing format that utilizes the 193rd bit as a framing bit, but whose Superframe is made up of 24 frames instead of 12 as in D4 format. ESF also provides CRC error detection and maintenance data link functions.

Exterior Gateway Protocol (EGP) - used by gateways in an internet, connecting autonomous networks.

Fairness - related to Generic Flow Control, fairness is defined as meeting all of the agreed quality of service requirements by controlling the order of service for all active connections.

Far End Block Error (FEBE) - an error detected by extracting the 4-bit FEBE field from the path status byte (G1). The legal range for the 4-bit field is between 0000 and 1000, representing zero to eight errors. Any other value is interpreted as zero errors.

Far End Receive Failure (FERF) - a line error asserted when a 110 binary pattern is detected in bits 6, 7, 8 of the K2 byte for five consecutive frames. A line FERF is removed when any pattern other than 110 is detected in these bits for five consecutive frames.

Far-End - in a relationship between two devices in a circuit, the far-end device is the one that is remote.

Face Contact (FC) - Designation for fiber optic connector designed by Nippon Telegraph and Telephone which features a movable anti-rotation key allowing good repeatable performance despite numerous mating. Normally referred to as Fiber Connector, FC actually stands for Face Contact and sometimes linked with PC (Point Contact), designated as FC or FC-PC.

FCC Part 68 - The FCC rules regulating the direct connection of non-telephone company provided equipment to the public telephone network.

Federal Communications Commission (FCC) - a board of commissioners appointed by the President under the Communications Act of 1934, with the authority to regulate all interstate telecommunications originating in the United States, including transmission over phone lines.

Fiber Distributed Data Interface (FDDI) - high-speed data network that uses fiber-optic as the physical medium. Operates in similar manner to Ethernet or Token Ring, only faster.

File Transfer Protocol (FTP) - a TCP/IP protocol that lets a user on one computer access, and transfer data to and from, another computer over a network. ftp is usually the name of the program the user invokes to accomplish this task.

First-In, First-Out (FIFO) - method of coordinating the sequential flow of data through a buffer.

Flag - a bit pattern of six binary “1”s bounded by a binary “0” at each end (forms a 0111 1110 or Hex “7E”). It is used to mark the beginning and/or end of a frame.

Flow Control - The way in which information is controlled in a network to prevent loss of data when the receiving buffer is near its capacity.

***ForeThought* PNNI (FT-PNNI) -** a FORE Systems routing and signalling protocol that uses private ATM (NSAP) addresses; a precursor to ATM Forum PNNI (see PNNI).

Forward Error Correction (FEC) - A technique used by a receiver for correcting errors incurred in transmission over a communications channel without requiring retransmission of any information by the transmitter; typically involves a convolution of the transmitted bits and the appending of extra bits by both the receiver and transmitter using a common algorithm.

Forward Explicit Congestion Notification (FECN) - Bit set by a Frame Relay network to inform data terminal equipment (DTE) receiving the frame that congestion was experienced in the path from source to destination. DTE receiving frames with the FECN bit set can request that higher-level protocols take flow control action as appropriate.

Fractional T1 - the use of bandwidth in 64Kbps increments up to 1.544Mbps from a T1 facility.

Frame - a variable length group of data bits with a specific format containing flags at the beginning and end to provide demarcation.

Frame Check Sequence (FCS) - In bit-oriented protocols, a 16-bit field that contains transmission error checking information, usually appended to the end of the frame.

Frame Relay - a fast packet switching protocol based on the LAPD protocol of ISDN that performs routing and transfer with less overhead processing than X.25.

Frame Synchronization Error - an error in which one or more time slot framing bits are in error.

Frame-Based UNI (FUNI) - An ATM switch-based interface which accepts frame-based ATM traffic and converts it into cells.

Frame-Relay Service (FRS) - A connection oriented service that is capable of carrying up to 4096 bytes per frame.

Framing - a protocol that separates incoming bits into identifiable groups so that the receiving multiplexer recognizes the grouping.

Frequency Division Multiplexing (FDM) - a method of dividing an available frequency range into parts with each having enough bandwidth to carry one channel.

Gbps - gigabits per second (billion)

Generic Cell Rate Algorithm (GCRA) - an algorithm which is employed in traffic policing and is part of the user/network service contract. The GCRA is a scheduling algorithm which ensures that cells are marked as conforming when they arrive when expected or later than expected and non-conforming when they arrive sooner than expected.

Generic Connection Admission Control (GCAC) - This is a process to determine if a link has potentially enough resources to support a connection.

Generic Flow Control (GFC) - the first four bits of the first byte in an ATM cell header. Used to control the flow of traffic across the User-to-Network Interface (UNI), and thus into the network. Exact mechanisms for flow control are still under investigation and no explicit definition for this field exists at this time. (This field is used only at the UNI; for NNI-NNI use (between network nodes), these four bits provide additional network address capacity, and are appended to the VPI field.)

GIO - a proprietary bus architecture used in certain Silicon Graphics, Inc. workstations.

Header - protocol control information located at the beginning of a protocol data unit.

Header Error Control (HEC) - a CRC code located in the last byte of an ATM cell header that is used for checking cell header integrity only.

High Density Bipolar (HDB3) - A bipolar coding method that does not allow more than 3 consecutive zeroes.

High Level Data Link Control (HDLC) - An ITU-TSS link layer protocol standard for point-to-point and multi-point communications.

High Performance Parallel Interface (HIPPI) - ANSI standard that extends the computer bus over fairly short distances at speeds of 800 and 1600 Mbps.

High-Speed Serial Interface (HSSI) - a serial communications connection that operates at speeds of up to 1.544 Mbps.

Host - In a network, the primary or controlling computer in a multiple computer installation.

HPUX - the Hewlett-Packard version of UNIX.

Hub - a device that connects several other devices, usually in a star topology.

I/O Module - FORE's interface cards for the LAX-20 LAN Access Switch, designed to connect Ethernet, Token Ring, and FDDI LANs to *ForeRunner* ATM networks.

Institute of Electrical and Electronics Engineers (IEEE) - the world's largest technical professional society. Based in the U.S., the IEEE sponsors technical conferences, symposia & local meetings worldwide, publishes nearly 25% of the world's technical papers in electrical, electronics & computer engineering, provides educational programs for members, and promotes standardization.

IEEE 802 - Standards for the interconnection of LAN computer equipment. Deals with the Data Link Layers of the ISO Reference Model for OSI.
IEEE 802.1 - Defines the high-level network interfaces such as architecture, internetworking and network management.
IEEE 802.2 - Defines the Logical Link Control interface between the Data Link and Network Layers.
IEEE 802.3 - Defines CSMA/CD (Ethernet).
IEEE 802.4 - Defines the token-passing bus.
IEEE 802.5 - Defines the Token Ring access methodology. This standard incorporates IBM's Token Ring specifications.
IEEE 802.6 - Defines Metropolitan Area Networks.
IEEE 802.7 - The broadband technical advisory group.
IEEE 802.8 - The fiber optics technical advisory group.
IEEE 802.9 - Defines integrated data and voice networks.

Integrated Services Digital Network (ISDN) - an emerging technology that is beginning to be offered by the telephone carriers of the world. ISDN combines voice and digital network services into a single medium or wire.

Interexchange Carriers (IXC) - Long-distance communications companies that provide service between Local Access Transport Areas (LATAs).

Interface Data - the unit of information transferred to/from the upper layer in a single interaction across a SAP. Each Interface Data Unit (IDU) controls interface information and may also contain the whole or part of the SDU.

Interface Data Unit (IDU) - The unit of information transferred to/from the upper layer in a single interaction across the SAP. Each IDU contains interface control information and may also contain the whole or part of the SDU.

Interim Local Management Interface (ILMI) - the standard that specifies the use of the Simple Network Management Protocol (SNMP) and an ATM management information base (MIB) to provide network status and configuration information.

Intermediate System (IS) - a system that provides forwarding functions or relaying functions or both for a specific ATM connection. OAM cells may be generated and received.

International Standards Organization (ISO) - a voluntary, non treaty organization founded in 1946 that is responsible for creating international standards in many areas, including computers and communications.

International Telephone and Telegraph Consultative Committee (CCITT) - the international standards body for telecommunications.

Internet - (note the capital "I") the largest internet in the world including large national backbone nets and many regional and local networks worldwide. The Internet uses the TCP/IP suite. Networks with only e-mail connectivity are not considered on the Internet.

internet - while an internet is a network, the term "internet" is usually used to refer to a collection of networks interconnected with routers.

Internet Addresses - the numbers used to identify hosts on an internet network. Internet host numbers are divided into two parts; the first is the network number and the second, or local, part is a host number on that particular network. There are also three classes of networks in the Internet, based on the number of hosts on a given network. Large networks are classified as Class A, having addresses in the range 1-126 and having a maximum of 16,387,064 hosts. Medium networks are classified as Class B, with addresses in the range 128-191 and with a maximum of 64,516 hosts. Small networks are classified as Class C, having addresses in the range 192-254 with a maximum of 254 hosts. Addresses are given as dotted decimal numbers in the following format:

nnn.nnn.nnn.nnn

In a Class A network, the first of the numbers is the network number, the last three numbers are the local host address.

In a Class B network, the first two numbers are the network, the last two are the local host address.

In a Class C network, the first three numbers are the network address, the last number is the local host address.

The following table summarizes the classes and sizes:

Class	First #	Max# Hosts
A	1-126	16,387,064
B	129-191	64,516
C	192-223	254

Network mask values are used to identify the network portion and the host portion of the address. Default network masks are as follows:

Class A - 255.0.0.0

Class B - 255.255.0.0

Class C - 255.255.255.0

Subnet masking is used when a portion of the host ID is used to identify a subnetwork. For example, if a portion of a Class B network address is used for a subnetwork, the mask could be set as 255.255.255.0. This would allow the third byte to be used as a subnetwork address. All hosts on the network would still use the IP address to get on the Internet.

Internet Control Message Protocol (ICMP) - the protocol that handles errors and control messages at the IP layer. ICMP is actually a part of the IP protocol layer. It can generate error messages, test packets, and informational messages related to IP.

Internet Engineering Task Force (IETF) - a large, open, international community of network designers, operators, vendors and researchers whose purpose is to coordinate the operation, management and evolution of the Internet to resolve short- and mid-range protocol and architectural issues.

Internet Protocol (IP) - a connectionless, best-effort packet switching protocol that offers a common layer over dissimilar networks.

Internetwork Packet Exchange (IPX) Protocol - a NetWare protocol similar to the Xerox Network Systems (XNS) protocol that provides datagram delivery of messages.

Interoperability - The ability of software and hardware on multiple machines, from multiple vendors, to communicate.

Interworking Function (IWF) - provides a means for two different technologies to interoperate.

IP Address - a unique 32-bit integer used to identify a device in an IP network. You will most commonly see IP addresses written in “dot” notation (e.g., 192.228.32.14).

IP Netmask - a 32-bit pattern that is combined with an IP address to determine which bits of an IP address denote the network number and which denote the host number. Netmasks are useful for sub-dividing IP networks. IP netmasks are written in “dot” notation (e.g., 255.255.0.0).

ISA Bus - a bus standard developed by IBM for expansion cards in the first IBM PC. The original bus supported a data path only 8 bits wide. IBM subsequently developed a 16-bit version for its AT class computers. The 16-bit AT ISA bus supports both 8- and 16-bit cards. The 8-bit bus is commonly called the PC/XT bus, and the 16-bit bus is called the AT bus.

Isochronous - signals carrying embedded timing information or signals that are dependent on uniform timing; usually associated with voice and/or video transmission.

International Telecommunications Union Telecommunications (ITU-T) - an international body of member countries whose task is to define recommendations and standards relating to the international telecommunications industry. The fundamental standards for ATM have been defined and published by the ITU-T (Previously CCITT).

J2 - Wide-area digital transmission scheme used predominantly in Japan that carries data at a rate of 6.312 Mbps.

Jitter - analog communication line distortion caused by variations of a signal from its reference timing position.

Joint Photographic Experts Group (JPEG) - An ISO Standards group that defines how to compress still pictures.

Jumper - a patch cable or wire used to establish a circuit, often temporarily, for testing or diagnostics; also, the devices, shorting blocks, used to connect adjacent exposed pins on a printed circuit board that control the functionality of the card.

Kbps - kilobits per second (thousand)

LAN Access Concentrator - a LAN access device that allows a shared transmission medium to accommodate more data sources than there are channels currently available within the transmission medium.

LAN Emulation Address Resolution Protocol (LE_ARP) - A message issued by a LE client to solicit the ATM address of another function.

LAN Emulation Client (LEC) - the component in an end system that performs data forwarding, address resolution, and other control functions when communicating with other components within an ELAN.

LAN Emulation Configuration Server (LECS) - the LECS is responsible for the initial configuration of LECs. It provides information about available ELANs that a LEC may join, together with the addresses of the LES and BUS associated with each ELAN.

LAN Emulation Server (LES) - the LES implements the control coordination function for an ELAN by registering and resolving MAC addresses to ATM addresses.

LAN Emulation (LANE) - technology that allows an ATM network to function as a LAN backbone. The ATM network must provide multicast and broadcast support, address mapping (MAC-to-ATM), SVC management, and a usable packet format. LANE also defines Ethernet and Token Ring ELANs.

lane - a program that provides control over the execution of the LAN Emulation Server (LES), Broadcast/Unknown Server (BUS), and LAN Emulation Configuration Server (LECS) on the local host.

Latency - The time interval between a network station seeking access to a transmission channel and that access being granted or received.

Layer Entity - an active layer within an element.

Layer Function - a part of the activity of the layer entities.

Layer Service - a capability of a layer and the layers beneath it that is provided to the upper layer entities at the boundary between that layer and the next higher layer.

Layer User Data - the information transferred between corresponding entities on behalf of the upper layer or layer management entities for which they are providing services.

le - a FORE program that implements both the LAN Emulation Server (LES) and the Broadcast/Unknown Server (BUS).

Leaky Bucket - informal cell policing term for the Generic Cell Rate Algorithm which in effect receives cells into a bucket and leaks them out at the specified or contracted rate (i.e., PCR).

Least Significant Bit (LSB) - lowest order bit in the binary representation of a numerical value.

lecs - a FORE program that implements the assignment of individual LECs to different emulated LANs.

leq - a FORE program that provides information about an ELAN. This information is obtained from the LES, and includes MAC addresses registered on the ELAN together with their corresponding ATM addresses.

Line Build Out (LBO) - Because T1 circuits require the last span to lose 15-22.5 dB, a selectable output attenuation is generally required of DTE equipment (typical selections include 0.0, 7.5 and 15 dB of loss at 772 KHz).

Line Code Violations (LCV) - Error Event. A Line Coding Violation (LCV) is the occurrence of either a Bipolar Violation (BPV) or Excessive Zeroes (EXZ) Error Event.

Link - An entity that defines a topological relationship (including available transport capacity) between two nodes in different subnetworks. Multiple links may exist between a pair of subnetworks. Synonymous with logical link.

Link Access Procedure, Balanced (LAPB) - Data link protocol in the X.25 protocol stack. LAPB is a bit-oriented protocol derived from HDLC. See also HDLC and X.25.

Link Down Trap - an SNMP trap, sent when an interface changes from a normal state to an error state, or is disconnected.

Link Layer - layer in the OSI model regarding transmission of data between network nodes.

Link Up Trap - an SNMP trap, sent when an interface changes from an error condition to a normal state.

Load Sharing - Two or more computers in a system that share the load during peak hours. During periods of non peak hours, one computer can manage the entire load with the other acting as a backup.

Local Access and Transport Area (LATA) - Geographic boundaries of the local telephone network, specified by the FCC, in which a single LEC may perform its operations. Communications outside or between LATAs are provided by IXCs.

Local Area Network (LAN) - a data network intended to serve an area of only a few square kilometers or less. Because the network is known to cover only a small area, optimizations can be made in the network signal protocols that permit higher data rates.

Logical Link Control (LLC) - protocol developed by the IEEE 802 committee for data-link-layer transmission control; the upper sublayer of the IEEE Layer 2 (OSI) protocol that complements the MAC protocol; IEEE standard 802.2; includes end-system addressing and error checking.

Loopback - a troubleshooting technique that returns a transmitted signal to its source so that the signal can be analyzed for errors. Typically, a loopback is set at various points in a line until the section of the line that is causing the problem is discovered.

looptest - program that tests an interface for basic cell reception and transmission functionality, usually used for diagnostic purposes to determine if an interface is functioning properly.

Loss Of Frame (LOF) - a type of transmission error that may occur in wide-area carrier lines.

Loss Of Pointer (LOP) - a type of transmission error that may occur in wide-area carrier lines.

Loss Of Signal (LOS) - a type of transmission error that may occur in wide-area carrier lines, or a condition declared when the DTE senses a loss of a DS1 signal from the CPE for more the 150 milliseconds (the DTE generally responds with an all ones “Blue or AIS” signal).

Management Information Base (MIB) - the set of parameters that an SNMP management station can query or set in the SNMP agent of a networked device (e.g., router).

Maximum Burst Size (MBS) - the Burst Tolerance (BT) is conveyed through the MBS which is coded as a number of cells. The BT together with the SCR and the GCRA determine the MBS that may be transmitted at the peak rate and still be in conformance with the GCRA.

Maximum Burst Tolerance - the largest burst of data that a network device is guaranteed to handle without discarding cells or packets. Bursts of data larger than the maximum burst size may be subject to discard.

Maximum Cell Delay Variance (MCDV) - This is the maximum two-point CDV objective across a link or node for the specified service category.

Maximum Cell Loss Ratio (MCLR) - This is the maximum ratio of the number of cells that do not make it across the link or node to the total number of cells arriving at the link or node.

Maximum Cell Transfer Delay (MCTD) - This is the sum of the fixed delay component across the link or node and MCDV.

Maximum Transmission Unit (MTU) - the largest unit of data that can be sent over a type of physical medium.

Mbps - megabits per second (million)

Media Access Control (MAC) - a media-specific access control protocol within IEEE 802 specifications; currently includes variations for Token Ring, token bus, and CSMA/CD; the lower sublayer of the IEEE’s link layer (OSI), which complements the Logical Link Control (LLC).

Media Attachment Unit (MAU) - device used in Ethernet and IEEE 802.3 networks that provides the interface between the AUI port of a station and the common medium of the Ethernet. The MAU, which can be built into a station or can be a separate device, performs physical layer functions including conversion of the digital data from the Ethernet interface, collision detection, and injection of bits onto the network.

Media Interface Connector (MIC) - fiber optic connector that joins fiber to the FDDI controller.

Message Identifier (MID) - message identifier used to associate ATM cells that carry segments from the same higher layer packet.

Metasignalling - an ATM Layer Management (LM) process that manages different types of signalling and possibly semipermanent virtual channels (VCs), including the assignment, removal, and checking of VCs.

Metasignalling VCs - the standardized VCs that convey metasignalling information across a User-to-Network Interface (UNI).

Metropolitan Area Network (MAN) - network designed to carry data over an area larger than a campus such as an entire city and its outlying area.

MicroChannel - a proprietary 16- or 32-bit bus developed by IBM for its PS/2 computers' internal expansion cards; also offered by others.

Minimum Cell Rate (MCR) - parameter defined by the ATM Forum for ATM traffic management, defined only for ABR transmissions and specifying the minimum value for the ACR.

Most Significant Bit (MSB) - highest order bit in the binary representation of a numerical value.

Motion Picture Experts Group (MPEG) - ISO group dealing with video and audio compression techniques and mechanisms for multiplexing and synchronizing various media streams.

MPOA Client - A device which implements the client side of one or more of the MPOA protocols, (i.e., is a SCP client and/or an RDP client. An MPOA Client is either an Edge Device Functional Group (EDFG) or a Host Behavior Functional Group (HBFG).

MPOA Server - An MPOA Server is any one of an ICFG or RSFG.

MPOA Service Area - The collection of server functions and their clients. A collection of physical devices consisting of an MPOA server plus the set of clients served by that server.

MPOA Target - A set of protocol address, path attributes, (e.g., internetwork layer QoS, other information derivable from received packet) describing the intended destination and its path attributes that MPOA devices may use as lookup keys.

Mu-Law - The PCM coding and companding standard used in Japan and North America.

Multicasting - The ability to broadcast messages to one node or a select group of nodes.

Multi-homed - a device having both an ATM and another network connection, like Ethernet.

Multimode Fiber Optic Cable (MMF) - fiber optic cable in which the signal or light propagates in multiple modes or paths. Since these paths may have varying lengths, a transmitted pulse of light may be received at different times and smeared to the point that pulses may interfere with surrounding pulses. This may cause the signal to be difficult or impossible to receive. This pulse dispersion sometimes limits the distance over which a MMF link can operate.

Multiplexing - a function within a layer that interleaves the information from multiple connections into one connection (see demultiplexing).

Multipoint Access - user access in which more than one terminal equipment (TE) is supported by a single network termination.

Multipoint-to-Multipoint Connection - a collection of associated ATM VC or VP links, and their associated endpoint nodes, with the following properties:

1. All N nodes in the connection, called Endpoints, serve as a Root Node in a Point-to-Multipoint connection to all of the (N-1) remaining endpoints.
2. Each of the endpoints can send information directly to any other endpoint, but the receiving endpoint cannot distinguish which of the endpoints is sending information without additional (e.g., higher layer) information.

Multipoint-to-Point Connection - a Point-to-Multipoint Connection may have zero bandwidth from the Root Node to the Leaf Nodes, and non-zero return bandwidth from the Leaf Nodes to the Root Node. Such a connection is also known as a Multipoint-to-Point Connection.

Multiprotocol over ATM (MPOA) - An effort taking place in the ATM Forum to standardize protocols for the purpose of running multiple network layer protocols over ATM.

Narrowband Channel - sub-voicegrade channel with a speed range of 100 to 200 bps.

National TV Standards Committee (NTSC) - Started in the US in 1953 from a specification laid down by the National Television Standards Committee. It takes the B-Y and R-Y color difference signals, attenuates them to I and Q, then modulates them using double-sideband suppressed subcarrier at 3.58MHz. The carrier reference is sent to the receiver as a burst during the back porch. An industry group that defines how television signals are encoded and transmitted in the US. (See also PAL, SECAM for non-U.S. countries).

Near-End - in a relationship between two devices in a circuit, the near-end device is the one that is local.

Network Layer - Layer three In the OSI model, the layer that is responsible for routing data across the network.

Network Management Entity (NM) - body of software in a switching system that provides the ability to manage the PNNI protocol. NM interacts with the PNNI protocol through the MIB.

Network Management Layer (NML) - an abstraction of the functions provided by systems which manage network elements on a collective basis, providing end-to-end network monitoring.

Network Management Station (NMS) - system responsible for managing a network or portion of a network by talking to network management agents, which reside in the managed nodes.

Network Module - ATM port interface cards which may be individually added to or removed from any *ForeRunner* ATM switch to provide a diverse choice of connection alternatives.

Network Parameter Control (NPC) - Defined as the set of actions taken by the network to monitor and control traffic from the NNI. Its main purpose is to protect network resources from malicious as well as unintentional misbehavior which can affect the QoS of other already established connections by detecting violations of negotiated parameters and taking appropriate actions. Refer to UPC.

Network Redundancy - Duplicated network equipment and/or data which can provide a backup in case of network failures.

Network Service Access Point (NSAP) - OSI generic standard for a network address consisting of 20 octets. ATM has specified E.164 for public network addressing and the NSAP address structure for private network addresses.

Network-to-Network Interface or Network Node Interface (NNI) - the interface between two public network pieces of equipment.

Node - A computer or other device when considered as part of a network.

Non Return to Zero (NRZ) - a binary encoding scheme in which ones and zeroes are represented by opposite and alternating high and low voltages and where there is no return to a zero (reference) voltage between encoded bits.

Non Return to Zero Inverted (NRZI) - A binary encoding scheme that inverts the signal on a “1” and leaves the signal unchanged for a “0”. (Also called transition encoding.)

Nonvolatile Storage - Memory storage that does not lose its contents when power is turned off.

NuBus - a high-speed bus used in Macintosh computers, structured so users can put a card into any slot on the board without creating conflict over the priority between those cards.

nx64K - This refers to a circuit bandwidth or speed provided by the aggregation of nx64 kbps channels (where n= integer > 1). The 64K or DS0 channel is the basic rate provided by the T Carrier systems.

Nyquist Theorem - In communications theory, a formula stating that two samples per cycle is sufficient to characterize a bandwidth limited analog signal; in other words, the sampling rate must be twice the highest frequency component of the signal (i.e., sample 4 KHz analog voice channels 8000 times per second).

Object Identifier (OID) - the address of a MIB variable.

Octet - a grouping of 8 bits; similar, but not identical to, a byte.

One’s Density - The requirement for digital transmission lines in the public switched telephone network that eight consecutive “0”s cannot be in a digital data stream; exists because repeaters and clocking devices within the network will lose timing after receiving eight “0”s in a row; a number of techniques are used to insert a “1” after every seventh-consecutive “0” (see Bit Stuffing).

Open Shortest Path First (OSPF) Protocol - a routing algorithm for IP that incorporates least-cost, equal-cost, and load balancing.

Open Systems Interconnection (OSI) - the 7-layer suite of protocols designed by ISO committees to be the international standard computer network architecture.

OpenView - Hewlett-Packard's network management software.

Operation and Maintenance (OAM) Cell - a cell that contains ATM LM information. It does not form part of the upper layer information transfer.

Optical Carrier level-n (OC-n) - The optical counterpart of STS-n (the basic rate of 51.84 Mbps on which SONET is based is referred to as OC-1 or STS-1).

Organizationally Unique Identifier (OUI) - Part of RFC 1483. A three-octet field in the SubNetwork Attachment Point (SNAP) header, identifying an organization which administers the meaning of the following two octet Protocol Identifier (PID) field in the SNAP header. Together they identify a distinct routed or bridged protocol.

Out-of-Band Management - refers to switch configuration via the serial port or over Ethernet, not ATM.

Out-of-Frame (OOF) - a signal condition and alarm in which some or all framing bits are lost.

Packet - An arbitrary collection of data grouped and transmitted with its user identification over a shared facility.

Packet Assembler Disassembler (PAD) - interface device that buffers data sent to/from character mode devices, and assembles and disassembles the packets needed for X.25 operation.

Packet Internet Groper (ping) - a program used to test reachability of destinations by sending them an ICMP echo request and waiting for a reply.

Packet Level Protocol (PLP) - Network layer protocol in the X.25 protocol stack. Sometimes called X.25 Level 3 or X.25 Protocol.

Packet Switched Network (PSN) - a network designed to carry data in the form of packets. The packet and its format is internal to that network.

Packet Switching - a communications paradigm in which packets (messages) are individually routed between hosts with no previously established communications path.

Payload Scrambling - a technique that eliminates certain bit patterns that may occur within an ATM cell payload that could be misinterpreted by certain sensitive transmission equipment as an alarm condition.

Payload Type (PT) - bits 2...4 in the fourth byte of an ATM cell header. The PT indicates the type of information carried by the cell. At this time, values 0...3 are used to identify various types of user data, values 4 and 5 indicate management information, and values 6 and 7 are reserved for future use.

Peak Cell Rate - at the PHY Layer SAP of a point-to-point VCC, the Peak Cell Rate is the inverse of the minimum inter-arrival time T0 of the request to send an ATM-SDU.

Peak Cell Rate (PCR) - parameter defined by the ATM Forum for ATM traffic management. In CBR transmissions, PCR determines how often data samples are sent. In ABR transmissions, PCR determines the maximum value of the ACR.

Peer Entities - entities within the same layer.

Peripheral Component Interconnect (PCI) - a local-bus standard created by Intel.

Permanent Virtual Channel Connection (PVCC) - A Virtual Channel Connection (VCC) is an ATM connection where switching is performed on the VPI/VCI fields of each cell. A Permanent VCC is one which is provisioned through some network management function and left up indefinitely.

Permanent Virtual Circuit (or Channel) (PVC) - a circuit or channel through an ATM network provisioned by a carrier between two endpoints; used for dedicated long-term information transport between locations.

Permanent Virtual Path Connection (PVPC) - A Virtual Path Connection (VPC) is an ATM connection where switching is performed on the VPI field only of each cell. A PVPC is one which is provisioned through some network management function and left up indefinitely.

Phase Alternate Line (PAL) - Largely a German/British development in the late 60s, used in the UK and much of Europe. The B-Y and R-Y signals are weighted to U and V, then modulated onto a double-sideband suppressed subcarrier at 4.43MHz. The V (R-Y) signal's phase is turned through 180 degrees on each alternate line. This gets rid of NTSC’s hue changes with phase errors at the expense of de-saturation. The carrier reference is sent as a burst in the back porch. The phase of the burst is alternated every line to convey the phase switching of the V signal. The burst's average phase is -V. (see NTSC for U.S.).

Physical Layer (PHY) - the actual cards, wires, and/or fiber-optic cabling used to connect computers, routers, and switches.

Physical Layer Connection - an association established by the PHY between two or more ATM-entities. A PHY connection consists of the concatenation of PHY links in order to provide an end-to-end transfer capability to PHY SAPs.

Physical Layer Convergence Protocol (PLCP) - a framing protocol that runs on top of the T1 or E1 framing protocol.

Physical Medium (PM) - Refers to the actual physical interfaces. Several interfaces are defined including STS-1, STS-3c, STS-12c, STM-1, STM-4, DS1, E1, DS2, E3, DS3, E4, FDDI-based, Fiber Channel-based, and STP. These range in speeds from 1.544Mbps through 622.08 Mbps.

Physical Medium Dependent (PMD) - a sublayer concerned with the bit transfer between two network nodes. It deals with wave shapes, timing recovery, line coding, and electro-optic conversions for fiber based links.

Plesiochronous - two signals are plesiochronous if their corresponding significant instants occur at nominally the same rate, with variations in rate constrained to specified limits.

Point of Demarcation - the dividing line between a carrier and the customer premise that is governed by strict standards that define the characteristics of the equipment on each side of the demarcation. Equipment on one side of the point of demarcation is the responsibility of the customer. Equipment on the other side of the point of demarcation is the responsibility of the carrier.

Point-to-Multipoint Connection - a collection of associated ATM VC or VP links, with associated endpoint nodes, with the following properties:

1. One ATM link, called the Root Link, serves as the root in a simple tree topology. When the Root node sends information, all of the remaining nodes on the connection, called Leaf nodes, receive copies of the information.

2. Each of the Leaf Nodes on the connection can send information directly to the Root Node. The Root Node cannot distinguish which Leaf is sending information without additional (higher layer) information. (See the following note for Phase 1.)

3. The Leaf Nodes cannot communicate directly to each other with this connection type.

Note: Phase 1 signalling does not support traffic sent from a Leaf to the Root.

Point-to-Point Connection - a connection with only two endpoints.

Point-to-Point Protocol (PPP) - Provides a method for transmitting packets over serial point-to-point links.

Policing - the function that ensures that a network device does not accept traffic that exceeds the configured bandwidth of a connection.

Port Identifier - The identifier assigned by a logical node to represent the point of attachment of a link to that node.

Presentation Layer - Sixth layer of the OSI model, providing services to the application layer.

Primary Reference Source (PRS) - Equipment that provides a timing signal whose long-term accuracy is maintained at 1 x 10 -11 or better with verification to universal coordinated time (UTC) and whose timing signal is used as the basis of reference for the control of other clocks within a network.

Primitive - an abstract, implementation-independent interaction between a layer service user and a layer service provider.

Priority - the parameter of ATM connections that determines the order in which they are reduced from the peak cell rate to the sustained cell rate in times of congestion. Connections with lower priority (4 is low, 1 is high) are reduced first.

Private Branch Exchange (PBX) - a private phone system (switch) that connects to the public telephone network and offers in-house connectivity. To reach an outside line, the user must dial a digit like 8 or 9.

Private Network Node Interface or Private Network-to-Network Interface (PNNI) - a protocol that defines the interaction of private ATM switches or groups of private ATM switches

Programmable Read-Only Memory (PROM) - a chip-based information storage area that can be recorded by an operator but erased only through a physical process.

Protocol - a set of rules and formats (semantic and syntactic) that determines the communication behavior of layer entities in the performance of the layer functions.

Protocol Control Information - the information exchanged between corresponding entities using a lower layer connection to coordinate their joint operation.

Protocol Data Unit (PDU) - a unit of data specified in a layer protocol and consisting of protocol control information and layer user data.

Proxy - the process in which one system acts for another system to answer protocol requests.

Proxy Agent - an agent that queries on behalf of the manager, used to monitor objects that are not directly manageable.

Public Data Network (PDN) - a network designed primarily for data transmission and intended for sharing by many users from many organizations.

Pulse Code Modulation (PCM) - a modulation scheme that samples the information signals and transmits a series of coded pulses to represent the data.

Q.2931 - Derived from Q.93B, the narrowband ISDN signalling protocol, an ITU standard describing the signalling protocol to be used by switched virtual circuits on ATM LANs.

Quality of Service (QoS) - Quality of Service is defined on an end-to-end basis in terms of the following attributes of the end-to-end ATM connection:

Cell Loss Ratio
Cell Transfer Delay
Cell Delay Variation

Queuing Delay (QD) - refers to the delay imposed on a cell by its having to be buffered because of unavailability of resources to pass the cell onto the next network function or element. This buffering could be a result of oversubscription of a physical link, or due to a connection of higher priority or tighter service constraints getting the resource of the physical link.

Radio Frequency Interference (RFI) - the unintentional transmission of radio signals. Computer equipment and wiring can both generate and receive RFI.

Real-Time Clock - a clock that maintains the time of day, in contrast to a clock that is used to time the electrical pulses on a circuit.

Red Alarm - In T1, a red alarm is generated for a locally detected failure such as when a condition like OOF exists for 2.5 seconds, causing a CGA, (Carrier Group Alarm).

Reduced Instruction Set Computer (RISC) - a generic name for CPUs that use a simpler instruction set than more traditional designs.

Redundancy - In a data transmission, the fragments of characters and bits that can be eliminated with no loss of information.

Registration - The address registration function is the mechanism by which Clients provide address information to the LAN Emulation Server.

Relaying - a function of a layer by means of which a layer entity receives data from a corresponding entity and transmits it to another corresponding entity.

Request To Send (RTS) - an RS-232 modem interface signal (sent from the DTE to the modem on pin 4) which indicates that the DTE has data to transmit.

Requests For Comment (RFCs) - IETF documents suggesting protocols and policies of the Internet, inviting comments as to the quality and validity of those policies. These comments are collected and analyzed by the IETF in order to finalize Internet standards.

RFC1483 - Multiprotocol Encapsulation over ATM Adaptation Layer 5.
RFC1490 - Multiprotocol Interconnect over Frame Relay.
RFC1577 - Classical IP and ARP over ATM.
RFC1755 - ATM Signaling Support for IP over ATM.

Robbed-Bit Signaling - In T1, refers to the use of the least significant bit of every word of frames 6 and 12 (D4), or 6, 12, 18, and 24 (ESF) for signaling purposes.

Route Server - A physical device that runs one or more network layer routing protocols, and which uses a route query protocol in order to provide network layer routing forwarding descriptions to clients.

Router - a device that forwards traffic between networks or subnetworks based on network layer information.

Routing Domain (RD) - A group of topologically contiguous systems which are running one instance of routing.

Routing Information Protocol (RIP) - a distance vector-based protocol that provides a measure of distance, or hops, from a transmitting workstation to a receiving workstation.

Routing Protocol - A general term indicating a protocol run between routers and/or route servers in order to exchange information used to allow computation of routes. The result of the routing computation will be one or more forwarding descriptions.

SBus - hardware interface for add-in boards in later-version Sun 3 workstations.

Scalable Processor Architecture Reduced instruction set Computer (SPARC) - a powerful workstation similar to a reduced-instruction-set-computing (RISC) workstation.

Segment - a single ATM link or group of interconnected ATM links of an ATM connection.

Segmentation And Reassembly (SAR) - the SAR accepts PDUs from the CS and divides them into very small segments (44 bytes long). If the CS-PDU is less than 44 bytes, it is padded to 44 with zeroes. A two-byte header and trailer are added to this basic segment. The header identifies the message type (beginning, end, continuation, or single) and contains sequence numbering and message identification. The trailer gives the SAR-PDU payload length, exclusive of pad, and contains a CRC check to ensure the SAR-PDU integrity. The result is a 48-byte PDU that fits into the payload field of an ATM cell.

Selector (SEL) - A subfield carried in SETUP message part of ATM endpoint address Domain specific Part (DSP) defined by ISO 10589, not used for ATM network routing, used by ATM end systems only.

Semipermanent Connection - a connection established via a service order or via network management.

Serial Line IP (SLIP) - A protocol used to run IP over serial lines, such as telephone circuits or RS-232 cables, interconnecting two systems.

Service Access Point (SAP) - the point at which an entity of a layer provides services to its LM entity or to an entity of the next higher layer.

Service Data Unit (SDU) - a unit of interface information whose identity is preserved from one end of a layer connection to the other.

Service Specific Connection Oriented Protocol (SSCOP) - an adaptation layer protocol defined in ITU-T Specification: Q.2110.

Service Specific Convergence Sublayer (SSCS) - The portion of the convergence sublayer that is dependent upon the type of traffic that is being converted.

Session Layer - Layer 5 in the OSI model that is responsible for establishing and managing sessions between the application programs running in different nodes.

Severely Errored Seconds (SES) - a second during which more event errors have occurred than the SES threshold (normally 10-3).

Shaping Descriptor - *n* ordered pairs of GCRA parameters (I,L) used to define the negotiated traffic shape of an APP connection. The traffic shape refers to the load-balancing of a network, where load-balancing means configuring data flows to maximize network efficiency.

Shielded Pair - Two insulated wires in a cable wrapped with metallic braid or foil to prevent interference and provide noise free transmission.

Shielded Twisted Pair (STP) - two or more insulated wires, twisted together and then wrapped in a cable with metallic braid or foil to prevent interference and offer noise-free transmissions.

Signaling System No. 7 (SS7) - The SS7 protocol has been specified by ITU-T and is a protocol for interexchange signaling.

Simple and Efficient Adaptation Layer (SEAL) - also called AAL 5, this ATM adaptation layer assumes that higher layer processes will provide error recovery, thereby simplifying the SAR portion of the adaptation layer. Using this AAL type packs all 48 bytes of an ATM cell information field with data. It also assumes that only one message is crossing the UNI at a time. That is, multiple end-users at one location cannot interleave messages on the same VC, but must queue them for sequential transmission.

Simple Gateway Management Protocol (SGMP) - the predecessor to SNMP.

Simple Mail Transfer Protocol (SMTP) - the Internet electronic mail protocol used to transfer electronic mail between hosts.

Simple Network Management Protocol (SNMP) - the Internet standard protocol for managing nodes on an IP network.

Simple Protocol for ATM Network Signalling (SPANS) - FORE Systems' proprietary signalling protocol used for establishing SVCs between FORE Systems equipment.

Single Mode Fiber (SMF) - Fiber optic cable in which the signal or light propagates in a single mode or path. Since all light follows the same path or travels the same distance, a transmitted pulse is not dispersed and does not interfere with adjacent pulses. SMF fibers can support longer distances and are limited mainly by the amount of attenuation. Refer to MMF.

Small Computer Systems Interface (SCSI) - a standard for a controller bus that connects hardware devices to their controllers on a computer bus, typically used in small systems.

Smart PVC (SPVC) - a generic term for any communications medium which is permanently provisioned at the end points, but switched in the middle. In ATM, there are two kinds of SPVCs: smart permanent virtual path connections (SPVPCs) and smart permanent virtual channel connections (SPVCCs).

snmpd - an SMNP agent for a given adapter card.

Source - Part of communications system which transmits information.

Source Address (SA) - The address from which the message or data originated.

Source MAC Address (SA) - A six octet value uniquely identifying an end point and which is sent in an IEEE LAN frame header to indicate source of frame.

Source Traffic Descriptor - a set of traffic parameters belonging to the ATM Traffic Descriptor used during the connection set-up to capture the intrinsic traffic characteristics of the connection requested by the source.

Spanning Tree Protocol - provides loop-free topology in a network environment where there are redundant paths.

Static Route - a route that is entered manually into the routing table.

Statistical Multiplexing - a technique for allowing multiple channels and paths to share the same link, typified by the ability to give the bandwidth of a temporarily idle channel to another channel.

Stick and Click (SC) - Designation for an Optical Connector featuring a 2.5 mm physically contacting ferrule with a push-pull mating design. Commonly referred to as Structured Cabling, Structured Connectors or Stick and Click

Stick and Turn (ST) - A fiber-optic connector designed by AT&T which uses the bayonet style coupling rather than screw-on as the SMA uses. The ST is generally considered the eventual replacement for the SMA type connector.

Store-and-Forward - the technique of receiving a message, storing it until the proper outgoing line is available, then retransmitting it, with no direct connection between incoming and outgoing lines.

Straight Tip (ST) - see *Stick and Turn*.

Structured Cabling (SC) - see *Stick and Click.*

Structured Connectors (SC) - see *Stick and Click*.

Sublayer - a logical subdivision of a layer.

SubNetwork Access Protocol (SNAP) - a specially reserved variant of IEEE 802.2 encoding SNAP indicates to look further into the packet where it will fine a Type field.

Subscriber Network Interface (SNI) - the interface between an SMDS end user's CPE and the network directly serving the end user, supported by either a DS1 or DS3 access arrangement.

Super Frame (SF) - a term used to describe the repeating 12 D4 frame format that composes a standard (non-ESF) T1 service.

Super User - a login ID that allows unlimited access to the full range of a device's functionality, including especially the ability to reconfigure the device and set passwords.

Sustainable Cell Rate (SCR) - ATM Forum parameter defined for traffic management. For VBR connections, SCR determines the long-term average cell rate that can be transmitted.

Sustained Information Rate (SIR) - In ATM this refers to the long-term average data transmission rate across the User-to-Network Interface. In SMDS this refers to the committed information rate (similar to CIR for Frame Relay Service).

Switch - Equipment used to interconnect lines and trunks.

Switched Connection - A connection established via signaling.

Switched Multimegabit Data Service (SMDS) - a high-speed, datagram-based, public data network service expected to be widely used by telephone companies in their data networks.

Switched Virtual Channel Connection (SVCC) - A Switched VCC is one which is established and taken down dynamically through control signaling. A Virtual Channel Connection (VCC) is an ATM connection where switching is performed on the VPI/VCI fields of each cell.

Switched Virtual Circuit (or Channel) (SVC) - a channel established on demand by network signalling, used for information transport between two locations and lasting only for the duration of the transfer; the datacom equivalent of a dialed telephone call.

Switched Virtual Path Connection (SVPC) - a connection which is established and taken down dynamically through control signaling. A Virtual Path Connection (VPC) is an ATM connection where switching is performed on the VPI field only of each cell.

Switching System - A set of one or more systems that act together and appear as a single switch for the purposes of PNNI routing.

Symmetric Connection - a connection with the same bandwidth specified for both directions.

Synchronous - signals that are sourced from the same timing reference and hence are identical in frequency.

Synchronous Data Link Control (SDLC) - IBM's data link protocol used in SNA networks.

Synchronous Optical Network (SONET) - a body of standards that defines all aspects of transporting and managing digital traffic over optical facilities in the public network.

Synchronous Payload Envelope (SPE) - the payload field plus a little overhead of a basic SONET signal.

Synchronous Transfer Mode (STM) - a transport and switching method that depends on information occurring in regular, fixed patterns with respect to a reference such as a frame pattern.

Synchronous Transport Signal (STS) - a SONET electrical signal rate.

Systeme En Coleur Avec Memoire (SECAM) - Sequential and Memory Color Television - Started in France in the late 60s, and used by other countries with a political affiliation. This is. The B-Y and R-Y signals are transmitted on alternate lines modulated on an FM subcarrier. The memory is a one line delay line in the receiver to make both color difference signals available at the same time on all lines. Due to FM, the signal is robust in difficult terrain.

Systems Network Architecture (SNA) - a proprietary networking architecture used by IBM and IBM-compatible mainframe computers.

T1 - a specification for a transmission line. The specification details the input and output characteristics and the bandwidth. T1 lines run at 1.544 Mbps and provide for 24 data channels. In common usage, the term “T1” is used interchangeably with “DS1.”

T1 Link - A wideband digital carrier facility used for transmission of digitized voice, digital data, and digitized image traffic. This link is composed of two twisted-wire pairs that can carry 24 digital channels, each operating at 64K bps at the aggregate rate of 1.544M bps, full duplex. Also referred to as DS-1.

T3 - a specification for a transmission line, the equivalent of 28 T1 lines. T3 lines run at 44.736 Mbps. In common usage, the term “T3” is used interchangeably with “DS3.”

Tachometer - in *ForeView*, the tachometer shows the level of activity on a given port. The number in the tachometer shows the value of a chosen parameter in percentage, with a colored bar providing a semi-logarithmic representation of that percentage.

Tagged Cell Rate (TCR) - An ABR service parameter, TCR limits the rate at which a source may send out-of-rate forward RM-cells. TCR is a constant fixed at 10 cells/second.

Telephony - The conversion of voices and other sounds into electrical signals which are then transmitted by telecommunications media.

Telnet - a TCP/IP protocol that defines a client/server mechanism for emulating directly-connected terminal connections.

Terminal Equipment (TE) - Terminal equipment represents the endpoint of ATM connection(s) and termination of the various protocols within the connection(s).

Throughput - Measurement of the total useful information processed or communicated by a computer during a specified time period, i.e. packets per second.

Time Division Multiplexing (TDM) - a method of traditional digital multiplexing in which a signal occupies a fixed, repetitive time slot within a higher-rate signal.

Token Ring - a network access method in which the stations circulate a token. Stations with data to send must have the token to transmit their data.

topology - a program that displays the topology of a FORE Systems ATM network. An updated topology can be periodically re-displayed by use of the interval command option.

Traffic - the calls being sent and received over a communications network. Also, the packets that are sent on a data network.

Traffic Management (TM) - The traffic control and congestion control procedures for ATM. ATM layer traffic control refers to the set of actions taken by the network to avoid congestion conditions. ATM layer congestion control refers to the set of actions taken by the network to minimize the intensity, spread and duration of congestion. The following functions form a framework for managing and controlling traffic and congestion in ATM networks and may be used in appropriate combinations:

Connection Admission Control
Feedback Control
Usage Parameter Control
Priority Control
Traffic Shaping
Network Resource Management
Frame Discard
ABR Flow Control

Traffic Parameter - A parameter for specifying a particular traffic aspect of a connection.

Trailer - the protocol control information located at the end of a PDU.

Transit Delay - the time difference between the instant at which the first bit of a PDU crosses one designated boundary, and the instant at which the last bit of the same PDU crosses a second designated boundary.

Transmission Control Protocol (TCP) - a specification for software that bundles and unbundles sent and received data into packets, manages the transmission of packets on a network, and checks for errors.

Transmission Control Protocol/Internet Protocol (TCP/IP) - a set of communications protocols that has evolved since the late 1970s, when it was first developed by the Department of Defense. Because programs supporting these protocols are available on so many different computer systems, they have become an excellent way to connect different types of computers over networks.

Transmission Convergence (TC) - generates and receives transmission frames and is responsible for all overhead associated with the transmission frame. The TC sublayer packages cells into the transmission frame.

Transmission Convergence Sublayer (TCS) - This is part of the ATM physical layer that defines how cells will be transmitted by the actual physical layer.

Transparent Asynchronous Transmitter/Receiver Interface (TAXI) - Encoding scheme used for FDDI LANs as well as for ATM; supports speed typical of 100 Mbps over multimode fiber.

Transport Layer - Layer Four of the OSI reference model that is responsible for maintaining reliable end-to-end communications across the network.

trap - a program interrupt mechanism that automatically updates the state of the network to remote network management hosts. The SNMP agent on the switch supports these SNMP traps.

Trivial File Transfer Protocol (TFTP) - Part of IP, a simplified version of FTP that allows files to be transferred from one computer to another over a network.

Twisted Pair - Insulated wire in which pairs are twisted together. Commonly used for telephone connections, and LANs because it is inexpensive.

Unassigned Cells - a generated cell identified by a standardized virtual path identifier (VPI) and virtual channel identifier (VCI) value, which does not carry information from an application using the ATM Layer service.

Unavailable Seconds (UAS) - a measurement of signal quality. Unavailable seconds start accruing when ten consecutive severely errored seconds occur.

UNI 3.0/3.1 - the User-to-Network Interface standard set forth by the ATM Forum that defines how private customer premise equipment interacts with private ATM switches.

Unicasting - The transmit operation of a single PDU by a source interface where the PDU reaches a single destination.

Universal Test & Operations Interface for ATM (UTOPIA) - Refers to an electrical interface between the TC and PMD sublayers of the PHY layer.

Unshielded Twisted Pair (UTP) - a cable that consists of two or more insulated conductors in which each pair of conductors are twisted around each other. There is no external protection and noise resistance comes solely from the twists.

Unspecified Bit Rate (UBR) - a type of traffic that is not considered time-critical (e.g., ARP messages, pure data), allocated whatever bandwidth is available at any given time. UBR traffic is given a “best effort” priority in an ATM network with no guarantee of successful transmission.

Uplink - Represents the connectivity from a border node to an upnode.

Usage Parameter Control (UPC) - mechanism that ensures that traffic on a given connection does not exceed the contracted bandwidth of the connection, responsible for policing or enforcement. UPC is sometimes confused with congestion management (see *congestion management*).

User Datagram Protocol (UDP) - the TCP/IP transaction protocol used for applications such as remote network management and name-service access; this lets users assign a name, such as “RVAX*2,S,” to a physical or numbered address.

User-to-Network Interface (UNI) - the physical and electrical demarcation point between the user and the public network service provider.

V.35 - ITU-T standard describing a synchronous, physical layer protocol used for communications between a network access device and a packet network. V.35 is most commonly used in the United States and Europe, and is recommended for speeds up to 48 Kbps.

Variable Bit Rate (VBR) - a type of traffic that, when sent over a network, is tolerant of delays and changes in the amount of bandwidth it is allocated (e.g., data applications).

Virtual Channel (or Circuit) (VC) - a communications path between two nodes identified by label rather than fixed physical path.

Virtual Channel Connection (VCC) - a unidirectional concatenation of VCLs that extends between the points where the ATM service users access the ATM Layer. The points at which the ATM cell payload is passed to, or received from, the users of the ATM Layer (i.e., a higher layer or ATMM-entity) for processing signify the endpoints of a VCC.

Virtual Channel Identifier (VCI) - the address or label of a VC; a value stored in a field in the ATM cell header that identifies an individual virtual channel to which the cell belongs. VCI values may be different for each data link hop of an ATM virtual connection.

Virtual Channel Link (VCL) - a means of unidirectional transport of ATM cells between the point where a VCI value is assigned and the point where that value is translated or removed.

Virtual Channel Switch - a network element that connects VCLs. It terminates VPCs and translates VCI values. The Virtual Channel Switch is directed by Control Plane functions and relays the cells of a VC.

Virtual Connection - an endpoint-to-endpoint connection in an ATM network. A virtual connection can be either a virtual path or a virtual channel.

Virtual Local Area Network (VLAN) - Work stations connected to an intelligent device which provides the capabilities to define LAN membership.

Virtual Network Software (VINES) - Banyan’s network operating system based on UNIX and its protocols.

Virtual Path (VP) - a unidirectional logical association or bundle of VCs.

Virtual Path Connection (VPC) - a concatenation of VPLs between virtual path terminators (VPTs). VPCs are unidirectional.

Virtual Path Identifier (VPI) - the address or label of a particular VP; a value stored in a field in the ATM cell header that identifies an individual virtual path to which the cell belongs. A virtual path may comprise multiple virtual channels.

Virtual Path Link (VPL) - a means of unidirectional transport of ATM cells between the point where a VPI value is assigned and the point where that value is translated or removed.

Virtual Path Switch - a network element that connects VPLs, it translates VPI (not VCI) values and is directed by Control Plane functions. The Virtual Path Switch relays the cells of a Virtual Path.

Virtual Path Terminator (VPT) - a system that unbundles the VCs of a VP for independent processing of each VC.

Virtual Private Data Network (VPDN) - a private data communications network built on public switching and transport facilities rather than dedicated leased facilities such as T1s.

Virtual Private Network (VPN) - a private voice communications network built on public switching and transport facilities rather than dedicated leased facilities such as T1s.

Virtual Source/Virtual Destination (VS/VD) - An ABR connection may be divided into two or more separately controlled ABR segments. Each ABR control segment, except the first, is sourced by a virtual source. A virtual source implements the behavior of an ABR source endpoint. Backwards RM-cells received by a virtual source are removed from the connection. Each ABR control segment, except the last, is terminated by a virtual destination. A virtual destination assumes the behavior of an ABR destination endpoint. Forward RM-cells received by a virtual destination are turned around and not forwarded to the next segment of the connection.

Virtual Tributary (VT) - a structure used to carry payloads such as DS1s that run at significantly lower rates than STS-1s.

Warm Start Trap - an SNMP trap which indicates that SNMP alarm messages or agents have been enabled.

Wide-Area Network (WAN) - a network that covers a large geographic area.

Wideband Channel - Communications channel with more capacity (19.2K bps) than the standard capacity of a voice grade line.

X.21 - ITU-T standard for serial communications over synchronous digital lines. The X.21 protocol is used primarily in Europe and Japan.

X.25 - a well-established data switching and transport method that relies on a significant amount of processing to ensure reliable transport over metallic media.

Yellow Alarm - An alarm signal sent back toward the source of a failed signal due to the presence of an AIS (may be used by APS equipment to initiate switching).

Zero Byte Time Slot Interchange (ZBTSI) - A technique used with the T carrier extended superframe format (ESF) in which an area in the ESF frame carries information about the location of all-zero bytes (eight consecutive “0”s) within the data stream.

Zero Code Suppression - The insertion of a “1” bit to prevent the transmission of eight or more consecutive “0” bits. Used primarily with T1 and related digital telephone company facilities, which require a minimum “1’s density” in order to keep the individual subchannels of a multiplexed, high speed facility active.

Zero-Bit Insertion - A technique used to achieve transparency in bit-oriented protocols. A zero is inserted into sequences of one bits that cause false flag direction.

Index

Index